Spring 3-9-2010

Service-Differentiated and Reliable Communication in Event-Based Wireless Sensor Networks

Yuyan Xue
University of Nebraska at Lincoln, sherry.xue@gmail.com

SERVICE-DIFFERENTIATED AND RELIABLE COMMUNICATION IN EVENT-BASED WIRELESS SENSOR NETWORKS

by

Yuyan Xue

A DISSERTATION

Presented to the Faculty of
The Graduate College at the University of Nebraska
In Partial Fulfillment of Requirements
For the Degree of Doctor of Philosophy

Major: Engineering
(Computer Engineering)

Under the Supervision of Professors Byrav Ramamurthy and Mehmet Can Vuran

Lincoln, Nebraska

May, 2010

SERVICE-DIFFERENTIATED AND RELIABLE COMMUNICATION IN EVENT-BASED WIRELESS SENSOR NETWORKS

Yuyan Xue, Ph.D.

University of Nebraska, 2010

Advisors: Byrav Ramamurthy and Mehmet Can Vuran

Wireless Sensor Networks (WSNs) consist of low-power embedded devices with integrated sensing, computing and wireless communication capabilities. These devices, called sensor nodes or motes, are often battery-powered and deployed in a distributed manner to provide observations on the physical world. Reliably and promptly collecting the sensing data to convey the features of a surveillance area, especially the events of interest, to the sink is one of the most critical requirements of WSN design. However, dynamic wireless channel conditions and the constrained energy resources make it a challenging task to provide the end-to-end performance guarantees in multi-hop WSNs.

The objective of this research is to develop new communication protocols that provide soft Quality of Service (QoS) guarantees for event-based WSNs in terms of latency, reliability and service-differentiation capability. By examining the application-specific end-to-end communication requirements and the fundamental resource limitations of WSNs, cross-layer solutions are developed in this work, to support Service-Differentiated Real-time Communication through an integrated MAC and network layer protocol, SDRCS, and to support Loss-Tolerant Reliable Event Sensing through a transport layer protocol, LTRES. An analytical framework based on a realistic log-normal channel model is also developed to quantitatively analyze how the end-to-end latency and energy efficiency can be improved by tuning the MAC and network layer protocol parameters. Besides the theoretical research, the design, implementation, and deployment details of a state-wide real-time

groundwater monitoring network in Nebraska are provided at the end, to demonstrate the advantages of wireless communication and networking technologies in improving the accuracy, coverage, and cost efficiency of real world environmental monitoring applications.

ACKNOWLEDGEMENTS

This dissertation is dedicated to all those who have extended their generous help during my Ph.D. study at the University of Nebraska-Lincoln.

First and foremost, I would like to express my sincere thanks to my advisor, Dr. Byrav Ramamurthy, for his invaluable support, guidance, and trust throughout my doctoral study. He led me into the world of computer networking and taught me how to conduct quality research. He reminded me to plan for a solid career path and encouraged me to work hard for it. His knowledge and patience helped me overcome many difficulties and guided me toward achieving my research and career goals.

My co-advisor, Dr. Mehmet Can Vuran, has always been available to listen and give advice. I am deeply grateful to him for the discussions between us that helped me refine the technical details of my work. His broad horizon of knowledge and infectious enthusiasm towards work benefited me a lot.

My Ph.D committee members Dr. Hong Jiang, Dr. Lisong Xu and Dr. Song Ci, have been giving me insightful advice and vital assistance on my graduate courses, research and future career. I would also like to thank them for carefully reading and commenting on my comprehensive exam report and dissertation manuscript.

I am also very grateful to all the members of the networking research group, with whom I have interacted during the course of my graduate study. Particularly, I would like to acknowledge Yong Wang, Jie Feng, Miao Wang and Srikanth Mallu for the many valuable discussions and collaborations.

My special thanks go to Dr. Mark Burbach, the PI of the groundwater monitoring project, and U.S. Department of Agriculture Risk Management Agency, for providing important financial support for my doctoral program through the project. The successful deployment of the state-wide groundwater monitoring network would not be possible without Dr. Burbach's dedicated work.

I am also grateful to the staff at UNL, for their various forms of support during my graduate study, especially to Shelley Everett, Sally Hawkins, Deb Heckens, LaRita Lang, and Eva Bachman.

The support and care from all my friends in Lincoln helped me overcome many setbacks and stay focused on my graduate study. I greatly value their friendship and deeply appreciate their belief in me. Finally, I would like to dedicate the dissertation to my family. Their constant care from a remote distance has been the source of my energy to achieve the goal.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless Sensor Networks (WSNs) consist of low-power embedded devices with integrated sensing, computation and wireless communication capabilities [1]. These devices, called sensor nodes or motes, are often battery-powered and deployed in a distributed manner to provide observations on the physical world. Based on the collective effort of a large number of sensor nodes, the sensing data can be delivered to a communication sink through multi-hop wireless links. Reliably and promptly collecting the sensing data to convey the features of a surveillance area, especially the events of interest, to the sink is one of the most critical requirements of WSN design. From the sink, the sensing data can be forwarded to a central data server for advanced data processing and permanent data storage. The end users, located anywhere in the world, then can monitor the surveillance area over the Internet. A typical multihop WSN architecture is shown in Fig. 1.1. WSNs enable a large variety of applications to monitor the physical world with reduced deployment cost and increased sensing scale and resolution. Example WSN applications include, but not limit to, environmental monitoring [2, 3], traffic control and vehicular networks [4, 5], battlefield surveillance [6], industrial production control [7], health care [8, 9] and structural monitoring [10].

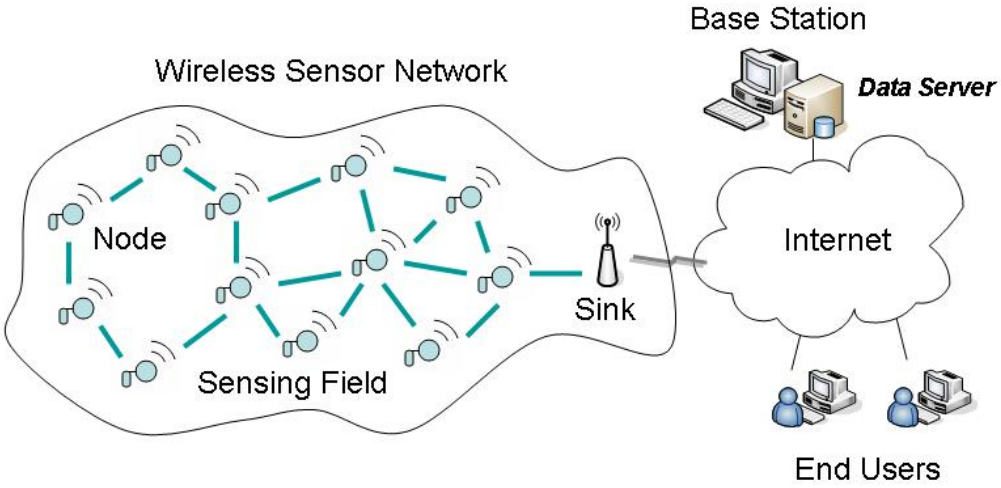


Figure 1.1: A multihop WSN architecture.

1.1 Research Challenges of QoS Provisioning in WSNs

Research on WSNs covers a wide ranges of topics, such as energy-efficient communication, network self-organization, data aggregation, node localization, topology control, and network security [11]. The advances in some of these topics have directly improved the performance of real-world WSN deployments, but major challenges remain. Among the biggest challenges is service-guaranteed Quality of Service (QoS) provisioning for mission-critical sensing applications. However, the unique characteristics of WSNs pose great challenges on service-differentiated real-time and reliable communication protocol design:

- *Unreliable and unpredictable wireless channel conditions*: WSNs utilize highly unreliable time-varying wireless channels affected by unpredictable environmental conditions. The channel quality can only be characterized by probabilistic models [12]; and the channel capacity may vary dramatically. The dynamic wireless channel conditions complicate the protocol design and performance modeling in WSNs.

- *Contention due to shared nature of wireless medium*: In a WSN, all sensor nodes that are located within the same interference area have to compete for the channel before

a packet transmission attempt. Only one of these nodes can acquire the channel and start the transmission. In addition, a node cannot successfully receive packets simultaneously sent by two or more nodes over the same channel, which introduces the hidden terminal problem. The complex contention scenario makes the bandwidth allocation and transmission scheduling become challenging tasks.

- *Limited energy budget*: Sensor nodes generally depend on batteries to provide non-renewable power supply. The limited energy budget calls for low-power operation, minimized traffic injection, and energy-efficient protocol design to prolong the network lifetime.

- *Dynamic network topology*: The sensor nodes are usually heterogeneous devices that are randomly deployed in the sensing field. The limited node lifetime, the time-varying channel conditions, and the duty-cycle operation increase the topology dynamics. No global topology information is available to a node. Thus, the communication protocol has to be designed in a distributed manner and adaptive to the topology changes.

- *Application-specific event sensing requirements*: A WSN can conduct multiple sensing tasks, which impose different latency and reliability requirements. Multiple events with different priority levels can also be detected in the same network. In addition, the latency or reliability requirements associated with a specific sensing task should be adjustable according to the dynamic physical environment. Therefore, the service-differentiation capability is necessary for QoS provisioning in WSNs.

1.2 Cross-Layer Design for Soft QoS Provisioning

The characteristics summarized in Section 1.1 depict the dynamic nature of WSN channel condition, network condition, and application requirements. Providing hard QoS guarantee (e.g., guaranteed packet receiving rate and end-to-end transmission delay for a specific data rate) in such a dynamic environment is almost impossible. In fact, many WSNs built today are ”best effort” systems, i.e., they provide few guarantees on the throughput, latency, or reliability of the communication [13, 14, 15]. As a result, systems have to be built first, then tuned later for specific end-to-end performance requirements; and there may still be transient periods of time when the QoS specification cannot be honored. This ad-hoc design methodology does not work well when we scale up the number of nodes in the WSN or when the WSN is used for mission-critical applications.

In this work, we try to follow a more structured design methodology by first examining the application-specific end-to-end latency and reliability requirements and then extrapolating these requirements into cross-layer protocol operation design. To this end, the end-to-end performance of the network can be predictable in a certain manner through the protocol control. Instead of providing hard QoS guarantees, we focus on how to provide soft QoS (or relaxed target QoS) guarantees in dynamic WSN environments, and make compromises on different latency or reliability objectives required by various sensing applications. Under this context, soft QoS, compared to the ”best effort systems”, is to improve the network resource utilization efficiency by cross-layer communication protocol control so that

- Application-specific end-to-end *latency* requirements can be guaranteed with higher packet delivery rate and lower energy consumption, and
- Application-specific end-to-end *reliability* requirements can be guaranteed with lower data rate, energy consumption and convergence time.

Soft QoS provisioning for end-to-end communication in WSNs requires collaborative

protocol control on multiple layers. From a layered perspective, power control in the physical layer affects the wireless interference range, therefore, changing the network topology and contention level. Transmission scheduling and channel contention management in the MAC layer affect the bandwidth allocation and utilization in the network. Routing or forwarding decisions, that are made in the network layer, control the flow distribution, therefore, affecting the achievable throughput and the local congestion level. Finally, the rate control in the transport layer affects the volume of traffic to be injected into the network, therefore, changing the network congestion level and the resource allocation among the outgoing flows.

The layered controls also potentially have mutual impacts because of the direct coupling among different layers [16]. Particularly, the physical layer can affect the MAC and routing decisions by adapting the transmission power. The MAC layer, that is responsible for scheduling and contention, determines the bandwidth utilization and the per-hop transmission delay, which then affects the decision at the network layer for route/link selection. The network layer chooses proper wireless links to forward packets to the destination, while the routing decision in turn changes the contention level at the MAC layer, and the interference observed at the physical layer. In addition, with fixed physical layer profile, the network QoS capacity determined by both MAC and network layer controls provide basic criteria for transport layer rate and admission control. Therefore, in order to develops efficient communication protocols for QoS provisioning in WSNs, the interdependencies among layers need to be characterized and exploited by cross-layer information exchange and joint optimization [16].

1.3 Research Objective and Our Contribution

In this work, we aim at soft QoS guarantees for end-to-end communication in WSNs. We consider a static WSN with heterogeneous sensor nodes, and a single sink. The nodes communicate through multihop wireless links, using a single channel and fixed transmission power. We consider an environmental monitoring application, where the sensing data is collected by the nodes and transmitted to the sink. Based on the data, the sink can monitor the sensing field and identify one or more areas of interest, where special events are predicted or detected. We call the area of interest as *event area*. According to the event urgency, the data packets can be assigned different end-to-end latency and reliability requirements. *To provide service-differentiated QoS guarantees for end-to-end event data communication and satisfy the application-specific latency or reliability requirements through MAC, network and transport layer protocol controls is the main objectives of this research.*

Two communication protocols, SDRCS (Service-Differentiated Real-time Communication Scheme) [17] and LTRES (Loss Tolerant Reliable Event Sensing protocol) [18, 19, 20], are developed to this end, focusing on soft real-time and reliability guarantees, respectively. SDRCS aims at providing soft real-time guarantees for event traffic with various end-to-end latency requirements through an integrated MAC and network layer design. As we discussed in Section 1.2, MAC and network layer controls are highly inter-dependent because scheduling policy, contention level, and forwarding decision all have strong mutual impacts. These controls also determine the QoS capacity of a WSN [16]. Therefore, SDRCS tries to exploit these inter-dependencies by combining the routing functionality with the RTS/CTS exchange procedure of a real-time MAC scheme to conduct receiver-contention-based packet forwarding. Accordingly, both the forwarding decision and forwarding operation can be designed based on local channel and network condition for packet

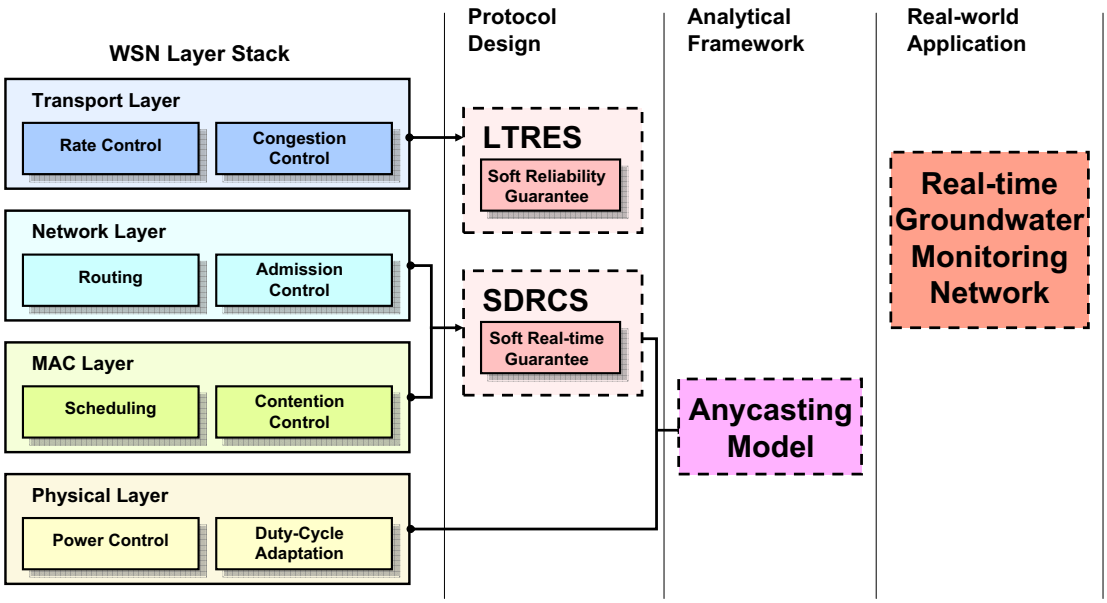


Figure 1.2: The contribution of this work in terms of WSN protocol design, analytical framework, and real-world sensor network application development.

traversal speed maximization. Meanwhile, the packet traversal speed estimation conducted at each sensor node takes charge of the traffic classification and admission control so that the soft real-time requirements can be guaranteed. The proposed SDRCS design requires minimum hardware support at the sensor nodes, where no localization, transmission power adaptation or multiple channel transmission support is required. It also adapts well to network dynamics, such as channel quality, local congestion and communication voids. The performance evaluation shows that compared with existing service-differentiated real-time communication schemes, for example, MMSpeed [21], SDRCS significantly improves the services-differentiation granularity for mixed-priority event traffic flows, and increases the on-time delivery rate by about 20% in unsynchronized WSNs with or without communication void. SDRCS also provides higher end-to-end throughput in terms of supporting higher source data rates with tight end-to-end latency requirements.

Compared with ”best-effort system”, SDRCS aims at providing better packet deliv-

ery rate for end-to-end event data transmissions subjected to specific deadline requirements. However, quantitatively characterizing the achievable end-to-end packet delivery rate using proposed receiver-contention-based forwarding (anycasting) scheme is vital for applying SDRCS into real-world event-based WSN applications. An analytical framework [22] is then provided to statistically analyze the performance of anycasting operation, and highlights the effect of physical layer power and duty-cycle control on the communication efficiency. Using a realistic log-normal channel model, we provide statistical end-to-end latency and energy analysis for anycasting operation, from which the probability of satisfying certain end-to-end latency and energy requirement for low-rate event traffic can be derived with a confidence level (statistical packet delivery rate). By exploring the relationship between the end-to-end latency and energy efficiency and the forwarding decision-independent anycasting parameters, two anycasting forwarding metrics are proposed for fully distributed forwarding decision. By exploring the relationship among the preamble length, the size of the forwarding set and the achievable end-to-end latency and energy efficiency, a series of preamble length control guidelines are proposed for low and extremely low duty-cycled WSNs. According to our analytical results and simulation validation, compared with the existing heuristic forwarding metrics, the proposed forwarding metrics help reduce the end-to-end latency and energy consumption by about 55% for anycasting with moderate preamble length. The proposed preamble length control guidelines help reduce, by more than half, the end-to-end energy and latency costs in low and extremely-low duty-cycled WSNs.

Next, a Loss Tolerant Reliable Event Sensing protocol, LTRES, is proposed to provide soft reliability guarantees for dynamic event traffic with various packet receiving rate (PRR) requirements. LTRES works as the transport layer complement of SDRCS to improve the network resource utilization and QoS through traffic volume control. The end-to-end event transport reliability is defined based on the PRR of the communication between an

identified event and the sink. A distributed source rate adaptation mechanism is proposed, incorporating a loss rate based congestion control mechanism, to regulate the event traffic injected into the network so that the reliability requirements can be satisfied under certain network capacity determined by MAC and network layer design. An equation based fair rate control algorithm is designed to improve the fairness among the traffic flows sharing the congestion path. The performance evaluations show that LTRES can provide event-based loss tolerant reliable data transport service for multiple events with short convergence time, low loss rate and high overall bandwidth utilization.

In addition to the theoretical research on QoS provisioning in WSNs, the design, implementation, and deployment details of a two-tier real-time environmental monitoring network infrastructure in Nebraska are provided at the end. As we discussed in Section 1.1, WSNs enable dynamic sensing tasks with high resolution and flexibility. However, the coverage of a WSN usually is limited to several square kilometers to ensure the reliable and real-time communication over multi-hop lossy wireless links. In order to implement large scale monitoring applications, for example, a state-wide groundwater monitoring network, a number of local WSNs need to be designed and deployed to cover local aquifer systems, which consist of the tier-two monitoring networks. The sensing data collected from local WSNs then need to be aggregated at the WSN sink and transmitted to a central base station through a tier-one long-haul wireless network. In our work, satellite communication technology has been used as the tier-one network for its high reliability, high throughput and low maintenance cost. With the proposed two-tier sensor network infrastructure, we demonstrate how the wireless communication and networking technologies help in improving the accuracy, flexibility, and cost efficiency of large-scale real world monitoring applications. Currently, the entire tier-one infrastructure has been designed and deployed to provide state-wide wireless coverage for 54 monitoring sites, where a tier-two WSN can be hooked up to improve the local sensing resolution. The communication protocol design

for tier-two WSN has been conducted through our theoretical research discussed in Chapter 2 to Chapter 4, and the deployment feasibility research is given in Section 5.3. The on-site WSN deployment at selected monitoring sites to provide improved local coverage and versatile sensing capability is planned in the near future. The complete two-tier sensor network infrastructure can serve as a real-world sensor network testbed to expedite the commercial adoption of WSN designs for large-scale environmental monitoring applications.

Our contribution in terms of WSN protocol design, analytical framework and real-world application development is summarized in Fig. 1.2.

1.4 Dissertation Organization

The rest of the dissertation is organized as follows. In Chapter 2, the design details and the performance evaluation of SDRCS are discussed, focusing on supporting service-differentiated real-time communication in event-based WSNs through integrated MAC and network controls. In Chapter 3, we provide an analytical framework to statistically analyze the performance of receiver-contention based forwarding scheme (or anycasting), which is proposed in SDRCS design, under a realistic channel model. In Chapter 4, the design details and the performance evaluation of LTRES protocol are provided, focusing on improving the network resource utilization and event data communication reliability through transport layer rate control. In Chapter 5, the design, implementation and deployment details of a two-tier real-time environmental monitoring network constructed in state of Nebraska are discussed. In Chapter 6, we conclude the work discussed in this dissertation and points out a number of interesting research points for future investigation.

Chapter 2

SDRCS: A MAC and Network Layer Integrated Real-time Communication Scheme for Event-Based WSNs

2.1 Introduction

Wireless Sensor Networks (WSNs) have emerged as a new generation of distributed embedded systems that provide observations on the physical world with low cost and high accuracy. Most of the WSN applications, such as battlefield surveillance [6], industrial production control [7], and structural monitoring [10], deal with various kinds of real-time constraints in response to the physical world [23, 24, 17]. In a typical real-time WSN application, as shown in Fig. 2.1, a number of sensor nodes are deployed to cover the sensing field. The predefined events can be detected by the nearby sensor nodes. The collected event information needs to be sent to the sink with certain deadline requirements so that the proper event response can be performed in a timely manner. According to the event urgency, the data packets associated with different events can be assigned different end-

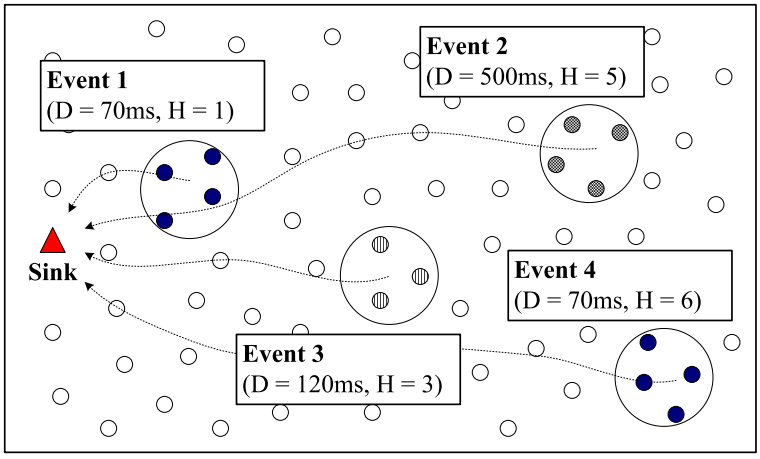


Figure 2.1: A service-differentiated real-time application in event-based WSNs.

to-end deadline requirements. Only the packets that are delivered to the sink before the deadline are deemed useful. Clearly, providing service-differentiated real-time guarantee for end-to-end data communication plays a vital role in expediting the widespread deployment of WSNs.

Providing real-time guarantee for end-to-end communication in WSNs is extremely challenging, compared with traditional networks, such as wireless LAN. First, WSNs require real-time provisioning for multi-hop communication over lossy channels. Because of the dynamic network and channel conditions en-route, only soft real-time guarantees can be provided. Second, the event-based traffic in WSNs may exhibit unpredictable spatial-temporal variations [23], based on which highly diverse real-time requirements can arise depending on different event locations and urgencies. As a result, traditional flow-based traffic classification methods, such as dividing QoS traffic into data, voice, video and control categories, may not be able to provide enough service-differentiation capability to the event traffic and ensure the prioritized transmission schedule. Third, the resource constraints restrict the design space of a feasible communication scheme for WSNs. For

example, location awareness and network synchronization may not be realistic assumptions for low-cost wireless sensor nodes. In addition, making the real-time communication scheme compatible with the duty-cycle design is vital to maximize the network lifetime for long-term monitoring application.

Supporting service-differentiated real-time communication in WSNs is a cross-layer task. First, an efficient prioritized medium access control (MAC) mechanism is required to provide service differentiation so that the packets with tighter deadline requirements can get higher priority to access the wireless channel and be delivered earlier to the destination. Some existing real-time communication protocols [25] use non-prioritized MAC design, such as B-MAC, with multiple priority queues to solve the in-node traffic prioritization. In this case, when a sender has multiple outgoing packets in queue, the packet with the tightest deadline requirements can be scheduled first for transmission. However, non-prioritized MAC with priority queue cannot resolve the inter-node traffic prioritization. When different senders within the same contention area try to send the packets with different deadline requirements, such MAC schemes cannot guarantee the packets with the highest priority to access the channel, and cause priority mismatch. Other recently proposed real-time communication protocols [21, 26] use dynamic Inter Frame Space (IFS) and Back-off Window (BW) extension-based Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) MAC scheme to resolve inter-node traffic prioritization. However, since IFS/BW extension based MAC schemes were originally designed for single-hop flow-based Wireless LAN (WLAN), it may experience severe bandwidth under-utilization, and service-differentiation degradation in multi-hop WSNs with diverse event-based real-time requirements. We will explain the limitations of these schemes in detail in Section 2.2.1.

Routing is another major challenge for real-time communication provisioning in WSNs. For efficient routing in multi-hop wireless networks, both the end-to-end hop count and the per-hop transmission delay need to be considered so that a proper routing decision can

be made to minimize the end-to-end delay, thus improving the sustainable throughput and deadline constraints. In order to do that, most existing real-time routing schemes [26, 27, 21] assume precise location-awareness at sensor nodes and use table-based forwarding. Each sensor has to maintain a routing table with all neighboring nodes' location information and average pairwise one-hop transmission delay. Based on the table, the packets can be forwarded to the neighboring node to gain maximum forwarding speed. However, in WSNs, to enable the precise location awareness and to maintain the freshness of the routing table under dynamic channel conditions, a large number of control packets need to be exchanged even if no traffic is injected in the network. The significant control overhead introduced by table-based geographic routing greatly deteriorates the network life-time for event-based real-time applications.

Providing proper admission control for the traffic injected into the network is also important to improve the bandwidth-utilization and energy-efficiency of a real-time communication scheme in WSNs. By estimating the schedulability of the packet transmissions, a proper admission control policy should be applied to the outgoing traffic in a per-hop manner. As a result, the packet transmission that is unlikely to meet the required latency constraints should be rejected at an early stage of the end-to-end transmission. However, most existing WSN real-time communication schemes do not consider admission control or simply drop packets only when the end-to-end transmission deadline is missed.

In this chapter, a novel Service-Differentiated Real-time Communication Sche-me (SDRCS) is proposed to provide soft real-time guarantees for event-based converge-cast traffic in WSNs using a cross-layer design. Compared with the existing real-time communication schemes, the main contributions of SDRCS design are as follows.

Cross-layer real-time forwarding: SDRCS uses a dynamic forwarding technique to integrate the routing functionality with a CSMA/CA RTS/CTS based prioritized MAC scheme. In this way, a receiver contention process is performed at each hop of packet

forwarding, based on the proposed real-time forwarding metric. The neighboring nodes with better forwarding distance, lower traffic load and higher channel quality to satisfy the real-time requirement will receive a higher priority to forward the packet. No routing tables or neighboring node information need to be maintained or periodically exchanged for end-to-end communication and hence the control overhead is mitigated. Since the forwarding decision is made on-demand, SDRCS can adapt well to the network dynamics. More important, the fully distributed and on-demand forwarding design makes SDRCS suitable for duty-cycled WSN.

Efficient prioritized MAC design: To provide better service-differentiation capability for diverse end-to-end deadline requirements in WSN applications, a novel polling-contention-period-based prioritized MAC is proposed in SDRCS, instead of traditional IFS/BW extension-based MAC schemes. The proposed MAC design helps decrease the average IFS and BW sizes when the number of traffic priority categories is large, thus improving the overall bandwidth utilization of the end-to-end communication when 4 or more traffic priority categories are supported in the network.

Light-weight packet schedulability estimation: SDRCS includes a light-weight packet schedulability estimation mechanism through received signal strength (RSS)-based sensor node grouping technique and a uniform polling contention period design for traffic within any priority category. Based on the proposed packet schedulability estimation scheme, proper admission control and early deadline-miss packet drop policies are designed to prevent unschedulable packets from being injected into the network and degrading the bandwidth utilization.

The rest of the chapter is organized as follows: Section 2.2 discusses some existing solutions for real-time communication in WSNs, and points out their limitations. Section 2.3 provides an overview of SDRCS design, including assumptions, design goals and SDRCS components. Section 2.4 describes the SDRCS design details and protocol operations. Sec-

tion 2.5 provides extensive simulation to evaluate the performance of SDRCS in terms of average end-to-end latency and on-time delivery rate. The chapter is concluded in Section 2.6.

2.2 Related Work

In this section, we discuss some existing real-time communication schemes for wireless networks. Based on our discussion, we point out the limitations of the existing work and motivate our cross-layer SDRCS design.

Some solutions have been proposed for prioritized medium access control in wireless networks. These approaches can mainly be divided into two groups: reservation-based and contention-based [28]. Reservation-based schemes [29, 30, 31] usually use TDMA based MAC operation. A strict requirement of these schemes is that the sensor network needs to be accurately synchronized, which introduces high control overhead for WSN applications. In addition, the signaling period required by TDMA schemes decreases the bandwidth utilization with increasing control overhead especially under light traffic load.

Contention-based schemes usually use Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) with Request to Send/Clear to Send (RTS/CTS) based MAC operation. A prevalent approach for achieving prioritized MAC in CSMA/ CA design, such as in 802.11e or 802.11EDCA [32], is to dynamically adapt the the Inter Frame Space (IFS) and/or Back-off Window (BW) length according to different priority classes. Larger IFS is used to transmit the packets with lower priority level. While the number of packets with the same priority level increases, larger BW is used to resolve the collision. We refer to these prioritized MAC approaches as *Dynamic IFS/BW Extension* in general. Most recently proposed service-differentiated real-time communication schemes in WSNs [26, 21] use dynamic IFS/BW extension for differentiated MAC support and do not require network

synchronization, and thus are light-weight choices for low-cost WSN applications. However, they cannot work well when the number of supporting priority levels increases for diverse end-to-end deadline requirements in WSN applications. Since IFS/BW extension based MAC design tries to prioritize the medium access by increasing the IFS and BW size for lower priority traffic, the average IFS and BW size achieved in the end-to-end communication and the probability of priority reversion [28] dramatically increases, which will lead to a degraded overall bandwidth utilization. The revised version of 802.11 EDCA [32] limits the supported number of priority levels to 4 mainly because of this reason. However, on the other hand, if the number of supporting priority levels is limited to a small number, more traffic will be classified into the same priority level. Such a situation will not only lead to a degraded service differentiation ability but also an increased collision possibility in the medium access contention process. Since a higher collision possibility can introduce a larger average BW size, the average communication throughput will also be deteriorated. Based on the above discussion, fine service differentiation with minimized IFS/BW extension need to be supported in any efficient real-time MAC design for service-differentiated WSN applications.

Besides the prioritized MAC, different real-time routing approaches have been proposed in recent literature. Most of them [26, 27, 25, 21] use traditional table-based routing technique, where each sensor node maintains a routing table with all neighboring nodes listed. Based on different real-time routing metrics, one neighboring node that can satisfy the application-specific deadline requirement is selected as the next hop to complete the packet forwarding. RAP [26] uses a greedy geographic forwarding metric so that any outgoing packets are routed to the neighboring node with the shortest distance to the receiver. A significant limitation of the RAP design is that the greedy geographic forwarding does not consider the local network conditions, such as load-balance, congestion level and channel quality. Therefore, the RAP routing decision leads to unpredictable per-hop trans-

mission delay in dynamic WSN environments, which not only affects the communication throughput but also the packet traversal speed estimation. SPEED [27], RPAR [25] and MMSPEED [21] improve the real-time routing metric by considering both the geographic information and the average pairwise transmission delay of neighboring nodes. The pairwise transmission delay is usually affected by the local contention level, congestion level and channel quality. Using the location and delay information, the sender can evaluate the packet progress speed achieved by a neighboring node thus making the forwarding decision based on minimizing the end-to-end latency. However, the table-based real-time routing techniques encounter common limitations in WSNs. First, to maintain the freshness of the information listed in the routing table for optimized forwarding decision in dynamic WSNs, a number of control messages need to be exchange periodically at each sensor node, which introduces huge control overhead, especially for event-based WSN applications. Second, table-based routing techniques are not suitable for duty cycle design, which is, however, vital for energy conservation in WSNs. In an unsynchronized WSN, the sensor nodes with duty cycle design will randomly go to sleep mode to decrease the energy consumption. In this case, a table-based routing techniques cannot properly identify the active next-hop candidate.

In contrast to table-based forwarding techniques, a receiver-contention based dynamic forwarding technique has been proposed in recent studies [33, 34, 35, 36, 37] for multi-hop data communication in WSNs with minimal energy consumption. The routing functionality is combined with the CSMA/CA RTS/CTS based MAC design so that an adaptive receiver contention is performed at each hop. The sensor nodes with better forwarding distance, lower traffic load, higher channel quality or higher residual energy level can receive a higher priority to respond to the RTS packet with a CTS packet and thus become the next hop. No routing tables or neighboring node information need to be maintained or periodically exchanged. Since the forwarding decision is made on-demand, these schemes

can easily adapt to a distributed duty-cycle design. The existing dynamic forwarding techniques motivate the SDRCS design by allowing for an efficient cross-layer communication approach. However, since the existing dynamic forwarding approaches do not consider soft real-time provisioning in forwarding decision, new forwarding metrics based on prioritized MAC operations need to be designed so that the application-specific deadline requirements can be enforced in end-to-end packet forwarding.

Besides the aforementioned MAC and network layer solutions, some physical and transport layer protocols has been proposed recently to address energy conservation and reliable communication for latency constrained WSN applications. In the work proposed in [38], the authors found that the event detection probability and detection latency are functions of the duty-cycle of sensor nodes. Based on this observation, a distributed algorithm is proposed to regulate the active probability of sensor nodes so that an event occurring anywhere in the network can be detected by the sink with a maximum detection latency and a minimum detection probability. In the work proposed in [39], the authors found that the end-to-end communication reliability and latency achieved in event-based WSNs can be regulated through transport-layer rate control. By observing the average end-to-end communication delay and the on-time delivery rate at the sink, proper event data rate control mechanisms are applied to the sensor nodes within the event area so that the application-specific event transport reliability or latency requirements can be achieved at the sink. Since [38, 39] are independent of the MAC and network layer operations, the proposed SDRCS operations can be complemented with these works.

2.3 SDRCS Design Overview

2.3.1 Assumptions

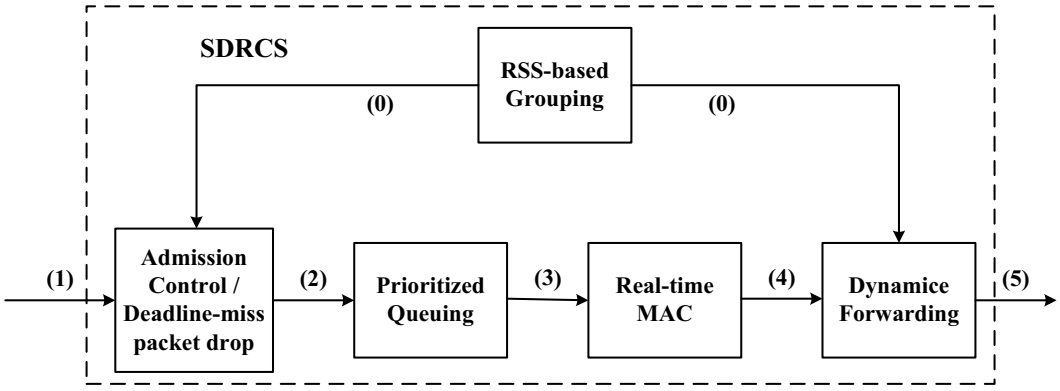
We consider homogenous multi-hop wireless sensor networks with a single sink (refer to Fig. 2.1). The sensor nodes are unsynchronized devices without location awareness. All the sensor nodes are configured with uniform transmission power. The sink and sensor nodes communicate using a single channel. The sensor nodes are capable of getting the received signal strength for each received packet. The above assumptions reflect the current hardware configurations of wireless sensor nodes [40].

We consider mission critical event sensing as our target application [23]. The predefined events are detected by the nearby sensor nodes and the event information should be converge-casted [41] to the sink. According to the event urgency, the data packets can be assigned different end-to-end deadline requirements. Only the packets delivered to the sink before the deadline are deemed useful. We also assume the networks to be connected, where at least one end-to-end forwarding path exists.

2.3.2 Design Goals

The main design goal of SDRCS is to support service-differentiated real-time communication for a WSN subject to the above assumptions. More specifically, our design satisfies the following objectives:

- **Service-differentiated soft real-time guarantee**: SDRCS should provide an accurate priority classification method and fine service differentiation granularity for dynamic event traffic with varying end-to-end deadline requirements. All packets arriving at the sink should be subject to the required end-to-end deadline. Proper admission control and early packet drop policy should be applied for achieving soft



(0) Group ID contains the end-to-end hop count information
(1) Incoming packet with an e2e deadline assignment
(2) Admitted packet with a per-hop deadline assignment
(3) Prioritized packet with a priority level assignment
(4) The highest-priority packet within the contention area obtains the channel
(5) The packet is forwarded to a next-hop who wins through the receiver-contension

Figure 2.2: Packet forwarding flow-chart within SDRCS main components at a sensor node.

real-time guarantees.

- **Minimum hardware support**: SDRCS should work well on the sensor nodes with highly constrained memory and energy resources. No expensive localization/synchronization devices, such as GPS, or complex algorithms should be required.

- **Adaptive to network dynamics**: SDRCS should adapt well to topology changes due to node failure or duty cycle design. Therefore, a fully distributed decision process is required for packet forwarding. It should also consider the dynamic channel quality and traffic load for packet delivery, thus adapting well to channel fading and network congestion.

2.3.3 SDRCS Main Components

In order to fulfill the design goals described in Section 2.3.2 for low-cost WSNs, we develop a real-time communication scheme consisting of five main components, as shown in Fig.

2.2:

- Received-Signal-Strength (RSS) based sensor node grouping
- Per-hop deadline based prioritized queueing
- Polling contention period based real-time medium access control (MAC)
- Receiver contention based dynamic forwarding
- Admission control and early-deadline-miss packet drop

The received signal strength (RSS) based sensor node grouping (Section 2.4.1) is designed for the post-deployment stage using limited broadcast. The grouping results can help the sensor nodes obtain end-to-end hop count estimation for better end-to-end latency estimation and accomplish routing decisions with low control overhead. With the grouping information, the prioritized queueing policy (Section 2.4.2) is designed to classify the incoming packet based on differentiated real-time requirements so that the packets with tighter deadline requirements can be transmitted earlier. The polling contention period based real-time MAC (Section 2.4.3) is then proposed to support prioritized channel access for the packets associated with different priority queues. Our real-time MAC is an improved design over dynamic IFS/BW extension based prioritized MAC design (see Section 2.2.1), which can help increase the service differentiation granularity with better overall bandwidth utilization. The receiver-contention based dynamic forwarding (Section 2.4.4) is embedded into the RTS/CTS exchanging process in real-time MAC for fully distributed on-demand routing. The real-time MAC and dynamic forwarding mechanism can guarantee that the packet with the highest priority level is delivered first to the best next-hop candidate so that the end-to-end latency can be minimized based on a local decision. We also estimate the schedulability of a packet transmission at each hop by providing admission control at the sender and early-deadline-miss packet drop policy enroute (Section 2.4.5). Thus, the

packet transmission that is unlikely to meet the required latency constraints is rejected at the early stage of the end-to-end transmission. We describe our proposed SDRCS scheme, and its four components, in detail in the next section.

2.4 SDRCS: A Service-Differentiated Real-time Communication Scheme for WSNs

In this section, the SDRCS design details are given in terms of the five SDRCS components. The protocol operations taken in each component are described. The relationship among different components and how a real-time packet is scheduled and forwarded by SDRCS design is shown in Fig. 2.2. The void avoidance capability of SDRCS design is discussed at the end of the section.

2.4.1 RSS-based Sensor Node Grouping

Most existing real-time communication protocols for WSNs assume precise location awareness at each sensor node [27] [21], which requires GPS equipment or complex localization schemes. In the absence of such precise location awareness, in our scheme, we use a Received Signal Strength (RSS) based sensor node grouping method to roughly strip the sensing field into layers, as shown in Fig. 2.3. The layer information can be used to estimate the hop-distance from the node to the sink, which enables the packet traversal speed estimation in the packet forwarding process. The accuracy of the hop-distance resulting from the grouping can be controlled by grouping granularity, defined as GRA. The basic grouping operations are given below:

(1) The sink initializes a *Grouping Message* broadcast with its group ID, where $G\_ID = 0$. (2) Each sensor node, that receives a grouping message with received signal strength

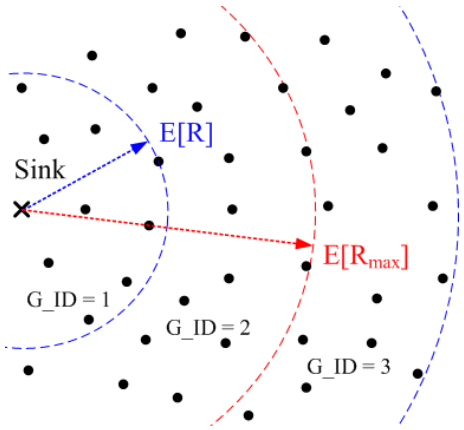


Figure 2.3: Received signal strength based sensor node grouping example with $GRA = 2$.

RSS higher than a pre-defined threshold RSS_{th} and does not have a group ID, assigns its group ID $G\_ID = G\_ID_r + 1$, where G_ID_r is the group ID value contained in the received grouping message. It then sets its back-off window as $BW = [G\_ID * slot, (G\_ID + 1) * slot]$, and broadcasts the grouping message containing its own group ID once. (3) Each sensor node, that receives a grouping message with received signal strength RSS lower than RSS_{th} and does not have a group ID, assigns its temporal group ID as $G\_ID_{temp} = G\_ID_r + GRA$. It then sets a timer that expires in GRA*Broadcast period. If a grouping message is received with received signal strength RSS higher than RSS_{th} before the timer expires and verifies that $G\_ID_{temp} > G\_ID_r + 1$, a sensor node will assign its group ID as $G\_ID = G\_ID_r + 1$. It then cancels the timer, sets its back-off window as $BW = [G\_ID * slot, (G\_ID + 1) * slot]$, and broadcasts the grouping message containing its own group ID once. (4) Each sensor node that has a G_ID_{temp} will assign its group ID as $G\_ID = G\_ID_{temp}$ when the timer expires. It then sets its back-off window as $BW = [G\_ID * slot, (G\_ID + 1) * slot]$, and broadcasts the grouping message containing its own group ID once. (5) The grouping broadcast stops when all the nodes finish their broadcast.

The grouping granularity is controlled by RSS_{th}. In our design, we considered log-

normal shadow channel model [12], where the RSS value at a receiver at a distance R from a transmitter is given by

$$RSS(R) = P_t - P_L(R_0) - 10\eta \log_{10}(\frac{R}{R_0}) + X_\sigma \ . \tag{2.1}$$

P_t is the transmit power in dBm, $P_L(R_0)$ is the path loss at a reference distance R_0 in dBm, η is the path loss exponent, and X_σ is the shadow fading component, where $X_\sigma \sim N(0, \sigma)$. With $RSS(R) = RSS_{th}$, the expected transmission range $E[R]$ of a broadcast message is given by

$$\begin{aligned} E[R] &= R_0 \cdot 10^{\frac{P_t - PL(R_0) - RSS_{th}}{10\eta}} \cdot E[10^{\frac{X_\sigma}{10\eta}}] & (2.2) \\ &= R_0 \cdot 10^{\frac{P_t - PL(R_0) - RSS_{th}}{10\eta}} \cdot e^{\frac{\sigma}{10\eta} \cdot lg10} \ . & (2.3) \end{aligned}$$

We also define the maximum transmission range $E[R_{max}]$ based on the noise power floor P_n, where

$$E[R_{max}] = R_0 \cdot 10^{\frac{P_t - PL(R_0) - P_n}{10\eta}} \cdot e^{\frac{\sigma}{10\eta} \cdot lg10} \ . \tag{2.4}$$

The grouping granularity, GRA, is then defined as

$$GRA = \frac{E[R_{max}]}{E[R]} = 10^{\frac{RSS_{th} - P_n}{10\eta}} \ . \tag{2.5}$$

By properly increasing RSS_{th}, the grouping granularity will be increased with more layers assigned to the network, which implies finer end-to-end hop-distance awareness at the sensor nodes. However, larger grouping granularity can also result in more grouping message broadcasts in the network with higher energy consumption. The back-off window, BW, design ensures that the sensor nodes with higher group ID cannot interrupt the *Grouping*

Message broadcast from a lower-group node so that the grouping process can be done in a layered manner.

The node grouping should be done at the post-deployment stage. After the RSS-based grouping process, the sensor nodes can be grouped into strip-style groups with $E[R]/GRA$ as the width of the strip. The density of the WSN will affect the grouping structure. With increasing node density, the result of grouping would approach perfect circular strips if the channel fading and noise components were homogeneous in the network [42]. The group ID can be used to estimate the hop-distance from the node to the sink and the packet forwarding can be guided towards the sink without precise location information. Similar approaches have been used for anchor beacon propagation in WSN localization, such as in [43] and [44]. However, in these approaches, the RSS information is not considered at the broadcast receivers as long as they are located within the transmission range. Therefore, the grouping granularity cannot be controlled in these existing approaches. The proposed grouping granularity control method can also be applied to other channel models, such as free space or two-ray model.

The main difference between using RSS-based grouping technique and traditional geographic localization technique in real-time communication scheme design lies in the node-to-sink distance definition. In a traditional geographic forwarding approach, the end-to-end distance is defined according to Euclidean distance; while in SDRCS design, the distance is defined as end-to-end hop count. As stated earlier, in a densely deployed WSN with homogeneous channel fading and noise components, the group ID can be a good indicator of node-to-sink geographic distance. However, in a sparsely deployed WSN or a WSN with dynamic channel fading and noise components, the group ID is more of an end-to-end hop count estimation, which may not be linearly related to geographic distance. Therefore, the proposed RSS-based grouping technique can help improve the accuracy of end-to-end hop-distance estimation in real WSN deployment, while avoiding the use of the expensive

precise localization schemes or devices as in [21] and [27].

The RSS-based grouping simulation results are shown in Section 2.5. According to the simulation results, by tuning the RSS_{th}, the granularity of grouping results can be properly adapted. In our simulation scenarios with the log-normal channel model and homogeneous shadow fading component, the RSS-based grouping design provides accurate enough end-to-end hop-distance information for the sensor node to make dynamic forwarding decisions.

2.4.2 Per-hop Deadline Based Prioritized Queueing Policy

In WSNs, an application-specific real-time requirement is usually presented as an end-to-end deadline, which indicates the maximum packet traversal time from the sender to the receiver [45]. However, in a multi-hop network, the end-to-end deadline is not the only criterion to determine the urgency of packet delivery. The end-to-end hop count also affects the packet delivery schedule. For example, if there are two schedulable packets with the same end-to-end deadline requirements competing for the channel, the one with higher end-to-end hop count should be scheduled first. If we assume that each sensor node is able to predict the end-to-end hop-count to the sink, the end-to-end deadline requirement can be broken down into a per-hop deadline requirement, L_{hop}^{Req},where

$$L_{hop}^{Req} = \frac{L_{e2e}}{HC_{e2e}} \, . \tag{2.6}$$

L_{e2e} is an application-specific parameter which reflects the required end-to-end delay for packet delivery. HC_{e2e} is the predicted hop-count value based on the G_ID of the sender, the GRA value and the forwarding strategy, which will be discussed in Section 2.4.4. L_{hop}^{Req} reflects the required per-hop traversal speed to achieve the end-to-end real-time guarantees in contention-based WSN. It can be used as an accurate enough indicator for packet delivery

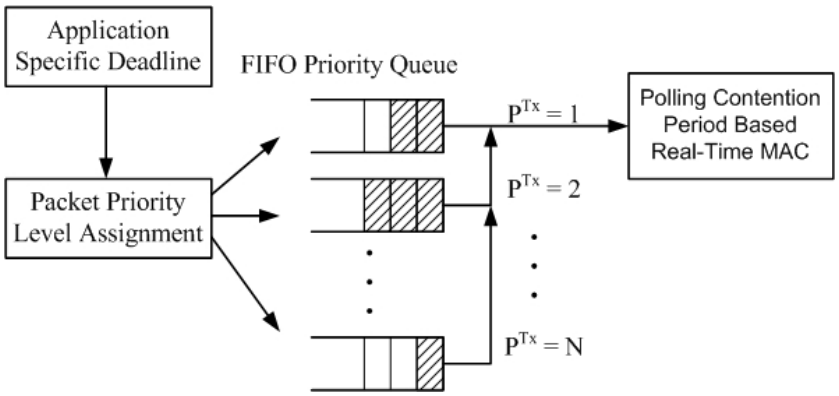


Figure 2.4: Per-hop deadline based priority queue at each sensor node for intra-node real-time traffic classification.

priority classification [45].

We use FIFO priority queue for packet scheduling at a node, as shown in Fig. 2.4. Since the prioritized MAC can only provide differentiated service for a limited number of priority classes, the per-hop deadline requirements are further mapped into N priority levels, where N is the number of the priority queues allocated at each sensor node.

In this chapter, we give a sample packet priority level P^{Tx} assignment policy as

$$P^{Tx} = min\left(\left\lfloor \frac{L_{hop}^{Req}}{L_{hop}^{Min}} \right\rfloor, N\right) . \tag{2.7}$$

The L_{hop}^{Min} is the minimum time required for one-hop packet forwarding, which depends on the MAC operations adopted in dynamic forwarding design. The L_{hop}^{Min} value for SDRCS is given in Section 2.4.4. Since Early Deadline First (EDF) has been proven as the most efficient scheduling policy for channel access in wireless networks [29], the packet with a larger priority level will be scheduled first for transmission.

Note that the given priority level assignment policy works well when the application-specific L_{hop}^{Req} is uniform distributed within its design space $[L_{hop}^{Min}, N * L_{hop}^{Min}]$. For different real-time applications with different L_{hop}^{Req} design spaces and distributions, different priority

level assignment policies can be used so that the incoming packets with various L_{hop}^{Req} values can be classified properly into N priority classes and placed into a associated priority queue for transmission [21].

2.4.3 Polling Contention Period-based Real-Time MAC

In order to better support the diverse end-to-end deadline requirements in WSN applications, we design a polling contention period based real-time MAC to support prioritized channel access.

As we mentioned in Section 2.2.1, Dynamic IFS/BW Extension is used by most existing real-time communication schemes for prioritized MAC support in WSNs. Such approaches employ extended Arbitrary Inter Frame Space (AIFS) and Back-off Window (BW) size for prioritized medium access contention. For a packet with priority level i, according to the IEEE 802.11EDCA [32], the AIFS value will be derived as follows :

$$AIFS_i = SIFS + i * SLOT\_TIME\ , \tag{2.8}$$

$$BW_i = (BW_1 + 1) * i - 1\ , \tag{2.9}$$

where SIFS is Short Inter Frames Space for controlling packet transmission contention. According to the dynamic IFS/BW extension based MAC design, the higher the number of priority levels supported in the network, the longer are the average AIFS and back-off window values, and the less the average throughput that can be achieved in the MAC operations.

In SDRCS, we use fixed number of polling slots instead of extended inter frame space and back-off window size for prioritized packet transmission contention, which is motivated by the bus access control mechanism in a computer system. The basic MAC operation adopted by SDRCS is shown in Fig. 2.5.

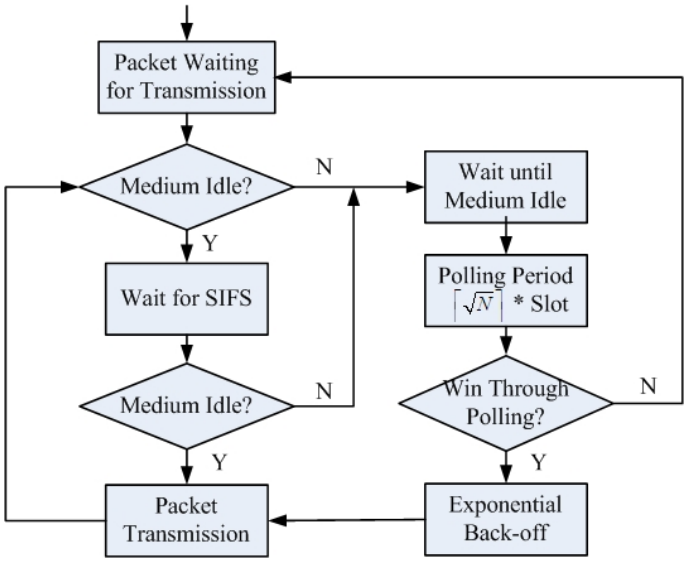


Figure 2.5: Polling period-based transmission contention in SDRCS Real-time MAC. The IFS and BW extension approach is replaced by $\lceil \sqrt{N} \rceil$ polling slots for all priority levels of traffic flows.

For any packet transmission, a sender first senses the medium status. If the medium is idle, the sender will wait for SIFS period of time and sense the medium again. If the medium remains idle, the sender will assume that no other packet is in transmission process and no other sender attempts to transmit a packet within the interference range at this time. The sender then initiates the packet transmission by sending out the RTS packet. If the medium is sensed to be busy after AIFS period of time, the sender will be notified that there is an on-going transmission within the interference range. It will wait until the medium is free.

Since several nodes within the interference range may have been waiting for this chance to transmit, all these nodes enter the polling period to compete for transmitting the RTS packet based on the priority level associated with the outgoing packet. The entire polling period consists of $\lceil \sqrt{N} \rceil$ polling slots for contention entities with N priority levels. For example, if 7 priority levels are supported in SDRCS design, 3 polling slots are required for medium access contention among all possible competitors within the interference area.

Table 2.1: Polling-slot Design for Maximum Priority Level =7

Priority Level	Slot 1	Slot 2	Slot 3
1	active	active	active
2	active	active	inactive
3	active	inactive	active
4	active	inactive	inactive
5	inactive	active	active
6	inactive	active	inactive
7	inactive	inactive	active

According to Table 2.1, any sensor node with an outgoing packet at priority level i will transmit a burst signal in its active polling slots and keep silent in its inactive polling slots. Any node that senses a burst in its inactive polling slots will be suppressed in the following transmission period. In this manner, only the node with the highest priority level among all competitors can enter the back-off period for RTS transmission. As a result, the number of competing nodes will dramatically decrease after the polling competition period and this results in less collision possibility; thus the BW can be set to a smaller size compared with that in dynamic IFS/BW extension based prioritized MAC design.

If we assume that all priority levels have the same amount of traffic load, with polling competition period design, our real-time MAC can result in better overall throughput when the number of priority levels satisfies $\left\lceil \sqrt{N} \right\rceil \leq \frac{N}{2} \Rightarrow N \geq 4$, without considering the throughput gain by the possible smaller BW.

2.4.4 Receiver Contention-based Dynamic Forwarding

Motivated by the cross-layer forwarding design discussed in Section 2.2, we propose receiver contention-based dynamic forwarding for converge-cast packet routing, which is combined into the RTS/CTS exchanging period of proposed real-time MAC design. According to the real-time MAC design in Section 2.4.3, if a sender i wins through a polling contention period and gains access to the medium after the exponential back-off period,

it will initiate an RTS transmission containing its own group ID, G_i. All the neighboring nodes that overhear this RTS message enter the receiver contention period. In the receiver contention period, only the sensor nodes with the same or lower group ID to G_i, become the qualified next-hop candidates so that the packet can only be forwarded towards the sink to gain non-negative packet traversal speed. The unqualified nodes enter the NAV (Network Allocation Vector) period. Each qualified next-hop candidate is required to evaluate its capability of maximizing the packet traversal speed for this transmission. The capability is classified into M priority levels for receiver contention.

Based on the above dynamic forwarding process, grouping granularity GRA gives the maximum number of groups that a packet can traverse within one hop. L_{hop}^{Min} gives the minimum time for one-hop packet transmission. Therefore, the maximum packet traversal speed that could be achieved by a forwarding decision without queuing delay is given by

$$Speed_{max} = \frac{GRA}{L_{hop}^{Min}} . \tag{2.10}$$

Regarding a specific next-hop candidate j, the average traversal speed $Speed_j$ by forwarding the packets to j is derived based on its average pairwise packet transmission time $t_{i,j}^{Avg}$, queue length L_Q, average per-packet queuing delay t_Q^{Avg} and group ID G_j, where

$$Speed_j = \frac{G_i - G_j}{t_{i,j}^{Avg} + L_Q * t_Q^{Avg}} . \tag{2.11}$$

In (2.11), the hop progress by a forwarding decision is given by the group ID difference between the sender and receiver $G_i - G_j$. The per-hop packet delivery delay consists of two parts, the packet transmission delay and packet queuing delay. For packet transmission delay, $t_{i,j}^{Avg}$ is calculated using a weighted moving average of the instantaneous packet

transmission time $t_{i,j}$ as

$$t_{i,j}^{Avg} = \alpha t_{i,j} + (1-\alpha) t_{i,j}^{Avg} \ . \tag{2.12}$$

For each packet transmission, the instantaneous packet transmission time $t_{i,j}$ is measured from the time an RTS is transmitted to the time of the corresponding ACK is received. If the packet is dropped due to exceeding the maximum retransmission times, N_{Re_Trans},

$$t_{i,j} = L_{hop}^{Min} * N_{Re\_Trans} \ .$$

In (2.12), $t_{i,j}^{Avg}$ is a good indicator of the link quality (packet error rate) of a potential receiver. A higher $t_{i,j}^{Avg}$ than L_{hop}^{Min} indicates the possible retransmission for a packet delivery between node i and j.

For packet queuing delay, the queue length L_Q reflects the traffic load at a particular sensor node. The average per-packet queuing delay t_Q^{Avg} reflects the local contention level for a particular next-hop candidate, which is calculated using a weighted moving average of the instantaneous per-packet queuing delay t_Q, where

$$t_Q^{Avg} = \beta t_Q + (1-\beta) t_Q^{Avg} \ .$$

The instantaneous per-packet queuing delay t_Q is measured from the time the last packet dequeue to the current packet dequeue for a certain priority level. Larger queuing length and per-packet queuing delay indicates lower packet traversal speed achieved by a forwarding decision.

Based on (2.10) and (2.11), a contention priority for a next-hop candidate j is given as

$$P^{Rx} = min\left(M - \left\lfloor \frac{Speed_j * M}{Speed_{max}} \right\rfloor, M\right) \ . \tag{2.13}$$

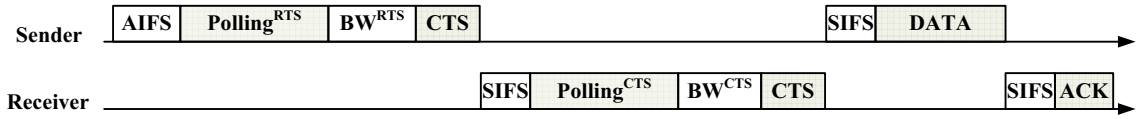


Figure 2.6: A complete prioritized packet transmission contention and receiver contention period for real-time MAC.

The above receiver priority assignment guarantees that

- The sensor node with a lower group ID will receive a higher priority for transmitting its CTS packet.
- For sensor nodes with the same group ID, the sensor node with a better channel quality and less traffic load will get a higher priority for transmitting its CTS packet.
- The packet cannot be forwarded to a sensor node with the same group ID as the sender to avoid loops in the route.
- The sensor node that can maximize the packet traversal speed towards sink will be assigned the highest priority.

Upon received the RTS packet and evaluating its forwarding priority based on (2.13), each possible next-hop candidate will first wait for an SIFS period and compete in $\left\lceil \sqrt{M} \right\rceil$ polling period according to its priority level, as the process introduced in Section 2.4.3 for RTS transmission. The winner enters an extra back-off period for possible collision among the candidates with the same contention priority or those that cannot hear the polling slots from each other. Since the collision probability is very low after the polling contention period, the back-off period is usually set to a much smaller number compared with BW_{RTS}. After the extra back-off period, the winning receiver sends back the CTS packet with its node ID to notify the sender. Accordingly, the sender will unicast the data packet to the winner, wait for the acknowledgement and finish the one-hop packet forwarding.

A complete prioritized packet transmission contention and receiver contention period for real-time MAC is shown in Fig. 2.6. According to our receiver-contention based dynamic forwarding operation, two important parameters are determined for (2.10) and (2.6). The minimum per-hop latency for the packet transmission with any priority level assignment, L_{hop}^{min}, can be given as

$$\begin{aligned} L_{hop}^{min} &= AIFS + t_{Polling}^{RTS} + \frac{1}{2}BW_{RTS,min} + t_{RTS} + \\ &\quad SIFS + t_{Data} + \\ &\quad SIFS + t_{Polling}^{CTS} + \frac{1}{2}BW_{CTS,min} + t_{CTS} + \\ &\quad SIFS + t_{ACK} \ , \end{aligned} \tag{2.14}$$

where $AIFS$ and $SIFS$ are arbitrary and short IFSs; $t_{Polling}^{RTS}$ and $t_{Polling}^{CTS}$ are fixed time of polling period for RTS and CTS packets; $BW_{RTS,min}$ and $BW_{CTS,min}$ are minimum back-off window values; $t_{RTS}, t_{CTS}, t_{Data}$ and t_{ACK} are RTS, CTS, Data and ACK packet transmission time respectively. The end-to-end hop count estimation for sender i is given by

$$HC_{e2e} = \frac{G_i}{Avg(G_{Fw}^i)} \ , \tag{2.15}$$

where $Avg(G_{Fw}^i)$ is the moving average of the number of groups a packet can traverse within a single hop transmission from node i and $0 \leq Avg(G_{Fw}^i) \leq GRA$. The receiver's group ID is obtained by i for each transmission through piggy-back in CTS packets. The initial value of $Avg(G_{Fw}^i)$ is set to GRA. The $Avg(G_{Fw}^i)$ value depends on the local network density, the channel quality and the network congestion level at the node i.

2.4.5 Admission Control and Early-Deadline-Miss Drop

Admission control is important in real-time provisioning for WSNs. Proper admission control policy can deny the unschedulable traffic entering the network and thus improve the bandwidth utilization and energy efficiency in end-to-end communication. According to our receiver-contention based dynamic forwarding design explained in Section 2.4.4, the minimum end-to-end deadline requirement for a packet initiated at sender i can be derived from (2.10) as

$$L_{e2e}^{min} = \frac{G_i * L_{hop}^{min}}{GRA} \ . \tag{2.16}$$

By using (2.15) in (2.7), any unschedulable end-to-end requirement will be mapped to a priority level i, where $1 \leq i \leq N$. Therefore, a simple admission control policy can be adopted at the sender, where all packets with priority level larger than N will not be admitted to the network.

We also design early-deadline-miss (EDM) drop policy for relaying nodes. For any relaying node k, the cumulative packet transmission time will be recorded as t_A and the remaining deadline for a packet, L_r will be calculated, where

$$L_r = L_{e2e} - t_A \ .$$

From (2.15), the updated per-hop deadline will be calculated at each relaying node based on

$$L_{hop}^{Req} = \frac{L_r}{G_k / Avg(G_{Fw}^k)} \ . \tag{2.17}$$

If the updated L_{hop}^{Req} is mapped into a priority level larger than N, the packet will be dropped because it is unlikely to be delivered to the sink on time based on the end-to-end hop count

estimation at node k. In contrast to the packet drop policy adopted by MMSpeed [21] or SPEED [27], which depends on the periodically updated per-hop delay information stored in the neighbor list, this early drop policy can better adapt to the dynamic channel and load conditions in WSNs, thus avoid false packet drops due to outdated per-hop pairwise delay information.

2.4.6 Void Avoidance

Many latency or energy sensitive communication schemes designed for WSNs, such as [26, 27, 21, 46], rely on a greedy forwarding strategy at every hop to transmit a data packet to a locally optimal next-hop node with a positive packet traversal progress towards the sink. However, such forwarding may not always be possible in common geographic forwarding design. For example, in a situation where all the neighboring nodes of a sender are of larger distance to the sink than the sender itself, the sender will fail to locate a qualified next-hop node, which has a positive geographic progress towards the sink. This undesirable phenomenon is usually called a communication void [47]. Avoiding communication void is a challenging problem for any geographic greedy forwarding approach. Although a dense deployment of wireless nodes can reduce the likelihood of the occurrence of a void in the network, it is still possible for some packets to encounter voids that are induced by dead nodes or the boundaries of a wireless network. These packets have to be discarded if only a greedy-forwarding strategy is used, even though a topologically valid path to the destination node may still exist. Thus, it is imperative to provide an effective and efficient void-handling approach in SDRCS design.

MMSpeed [21] and SPEED [27] use passive participation to deal with the communication voids. The idea of passive participation appears in [33] [48] and it exploits a self-healing property of network topology itself. Once a node identifies itself as a void node, it simply discards the data packet and keeps itself from forwarding any subsequent

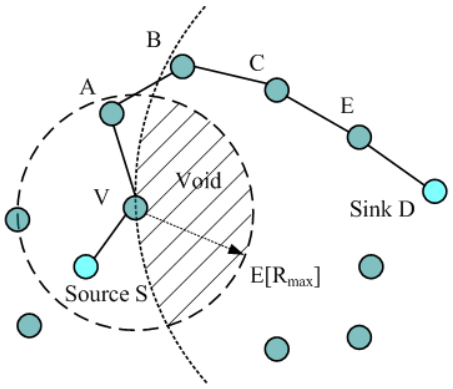


Figure 2.7: An example communication void where passive participation fails.

data packets toward the destination. The node may periodically check if it can locate a neighboring node with a positive progress in order to participate packet forwarding at a later time. This simple strategy has a reverse-propagation effect which eventually informs other intermediate nodes to explore other possible paths in the network, so that those nodes leading to a broken route can be avoided on routing paths. However, passive participation may not always be effective. For instance, as shown in Fig. 2.7, source node S wants to deliver a sequence of data packets to sink D. The first data packet is greedily forwarded to node V at the first hop. However, node V cannot continue to greedily forward the data packet. Node V drops the data packet and will not participate in forwarding the subsequent data packets for destination D any more. It seems to S that node V does not exist in the topology. However, no other node with a positive progress in its neighborhood can help forward the subsequent data packets. Thus, data packets will have to be discarded although a topologically valid path does exist from S to D: $S - V - A - B - C - E - D$. It was argued in [49] that passive participation is not effective in a randomly deployed wireless network with low density.

For SDRCS, communication void is handled inherently by grouping ID assignment and forwarding metric design. First note that the RSS-based grouping is a limited broadcast process initiated at the sink, as described in Section 2.4.1. Any node can be reached

by the the broadcast grouping message and assigned a group ID while the network is connected. In addition, any node with a group ID assignment must be able to reach the sink through the reversed broadcast path, if symmetric links are assumed between each pair of connected nodes. Then, considering the receiver-contention based dynamic forwarding operation described in Section 2.4.4, a packet forwarding fails to find a next hop only if one of the following condition is true:

- There is no node within the transmission range of the sender.
- Any node within the transmission range has a larger group ID than the sender.

However, RSS-based grouping design prevents either condition from occurring in SDRCS operation. First, since the network is assumed to be connected, there must be at least one node within the transmission range of the sender. Second, any sender must have at least one neighboring node with a smaller group ID, from which the grouping message is received. As a result, the SDRCS operation can guarantee that a packet can always be forwarded from the sender to a node with lower group ID and finally reach the sink, whose group ID equals to 0.

The main difference between group ID based forwarding and traditional geographic forwarding lies in the node-to-sink distance definition. In a traditional geographic forwarding approach, the distance is only defined in the geographic domain; while in group ID based forwarding, the distance is defined as end-to-end hop count. As we mentioned in Section 2.4.1, the group ID can be a good indicator of node-to-sink geographic distance in a densely deployed homogeneous WSN. However, in a sparsely deployed WSN with communication voids, the group ID is more of an end-to-end hop count estimation, which may not be linearly related to geographic distance. In Section 2.5.4, we will set up a simulation scenario to show how grouping results adapt to the communication voids and thus help the SDRCS design route around voids.

2.5 Performance Evaluation

The performance of our real-time communication scheme, SDRCS, is analyzed in GlomoSim [50] using the simulation parameters shown in Table 2.2. We choose log-normal shadow channel model [12] to reflect the channel dynamics in real WSN deployments and implement the model in the simulator. The node related parameters are also carefully chosen to reflect typical MicaZ node capabilities [40]. We conduct extensive simulation scenarios for SDRCS and compare its performance with the existing service differentiated real-time communication schemes, RAP [26] and MMSpeed [21]. The MAC operation parameters for dynamic IFS/BW extension based prioritized MAC design (used by RAP and MMSpeed), and real-time MAC (used by SDRCS), are listed in Table 2.3. For RAP and MMSpeed, AIFS[i], $BW_{RTS}^{Min}[i]$ and $BW_{RTS}^{Max}[i]$ values are defined based on the simulation setting in [21] and (2.8), where i is the data packet transmission priority level. For SDRCS, the L_{hop}^{min} value is derived according to (2.14).

Table 2.2: Simulation Environment Settings

Sensing field dimensions	(500×500) m
Sink location	$(25, 25)$
Number of sensor nodes	100
Node placement	Random uniform
Packet length	128 bytes
Radio bandwidth	250 kbps
Channel model	log-normal shadow
Path loss exponent	4
Shadow fading variance	6
Transmission power	1dBm
Noise power floor	-95dBm
Maximum transmission range	125m
Reference distance	0.3m

Two important end-to-end metrics are measured for real-time performance evaluation in our simulations:

Table 2.3: Dynamic IFS/BW extension based prioritized MAC and Real-time MAC parameters

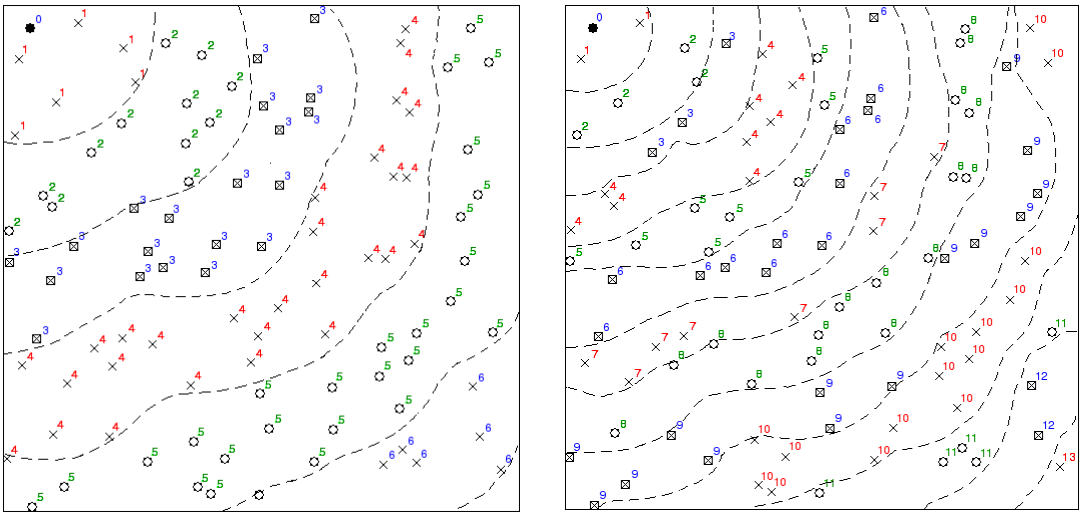
	MMSpeed & RAP	SDRCS
Retransmission Limit	7	7
Number of Priority Class	7	7
SIFS	10μs	10μs
Time Slot	20μs	20μs
AIFS[1]	$30 - 150\mu$s	80μs
BW_{RTS}^{Min}[1]	15 Slots	10 Slots
BW_{RTS}^{Max}[1]	255 Slots	200 Slots
BW_{CTS}	N/A	4 Slots
$L_{per-hop}^{min}$	N/A	2200μs

- **End-to-End On-Time Packet Delivery Rate**: The ratio of the number of unique packets received at the sink with end-to-end latency equal to or less than the end-to-end deadline requirement, to the total number of packets sent by the source node. This metric shows the end-to-end real-time capacity of the network achieved by a given communication scheme.

- **Average End-to-End Packet Transmission Latency**: Average end-to-end transmission time for all on-time delivered packets. The packets that are dropped enroute due to deadline-miss do not count for average end-to-end latency calculation. This metric shows the service-differentiation capability of a given communication scheme.

2.5.1 RSS-based Grouping with Varying Grouping Granularity

In this simulation scenario, we first examine the performance of RSS-based sensor node grouping technique. Fig. 2.8 (a) and (b) show a sample network topology generated for the simulation and the RSS-based geographic grouping results with $GRA = 1$ and $GRA = 2$ separately. From the simulation results, we can observe that in the sample network topology with homogeneous channel fading and noise components, simple RSS-based geographic grouping can properly divide the sensing field into near circular strips. Since the sensor



(a) Grouping results with $GRA = 1$ (b) Grouping results with $GRA = 2$

Figure 2.8: A sample simulation network topology with $node\ degree = 15$. The RSS-based group formation results with grouping granularity $GRA = 1$ and $GRA = 2$ are shown based on the sample topology. The number shown next to each node is the resulting group ID.

nodes distribution as well as the channel fading and noise components are homogeneous in the network, the node-to-sink hop count should be proportional to the node-to-sink distance. According to Fig. 2.9, the group ID assignment at the sensor node shows a near perfect linear relation to the average group-to-sink distance. By increasing the group granularity, GRA, the linear relation between the resulting group ID and the group-to-sink distance are refined with smaller error range.

Next we show how GRA affects the end-to-end performance of the SDRCS design. Two round of simulations are performed with different source nodes and end-to-end deadline requirements. Using the network topology shown in Fig. 2.8 (a), we choose the source nodes with the highest group ID to maximize the possible end-to-end hop count. In the first simulation round, one constant-bit-rate (CBR) event data flow CBR1 is generated from a node located at $(459, 411)$ and with end-to-end deadline requirement $L_{e2e}^{Req} = 30$ms. In the second round, another event data flow CBR2 is generated from a node located at $(402, 451)$ and with $L_{e2e}^{Req} = 60$ms. According to (2.7), $Priority_{CBR1}^{Tx} = 2$ and $Priority_{CBR2}^{Tx} = 4$. For

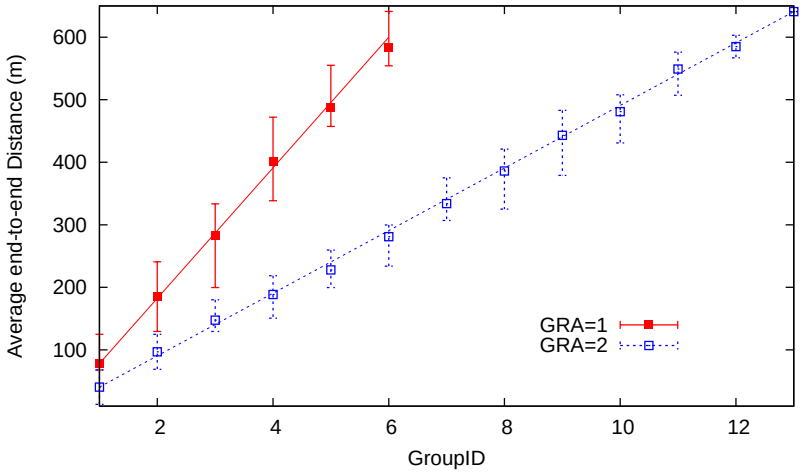
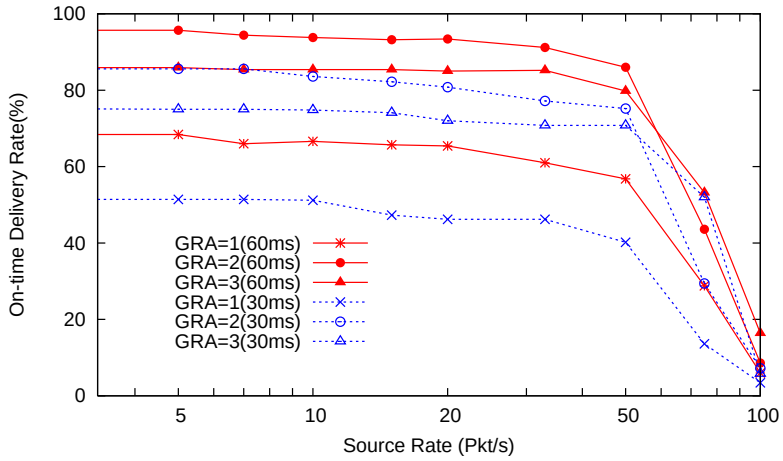


Figure 2.9: Average group to sink distance with $GRA = 1$ and $GRA = 2$ based on the sample network topology with $node\ degree = 15$.

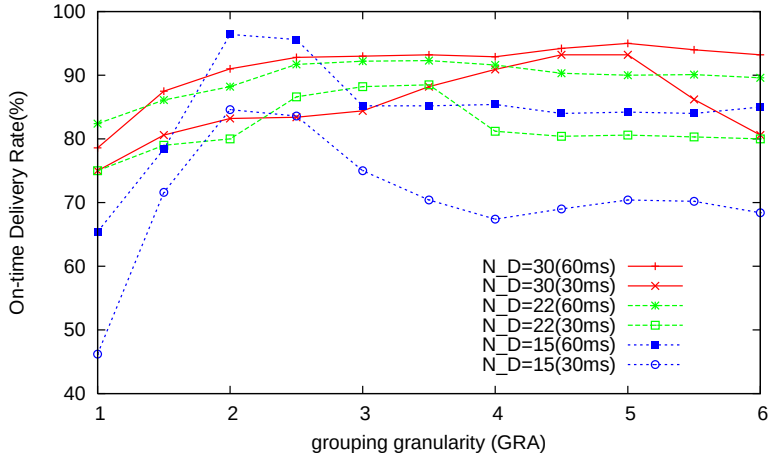
each round, 2000 packets are sent from the source node. The simulation is conducted 10 times with different random seeds and the average value collected from all 10 simulations.

Fig. 2.10 (a) shows the average on-time delivery rate of $CBR1$ and $CBR2$ with the GRA value varied from 1 to 3. From the simulation results, we can observe that increasing the grouping granularity, GRA, helps improve the end-to-end real-time performance for both priority levels of traffic because higher GRA implies better localization information. By choosing an proper GRA value, where $GRA = 2$ in the sample topology with $node\ degree = 15$, we can achieve up to 30% improvement on end-to-end on-time delivery rate for both traffic flows. However, increasing GRA without considering the network density will lead to uneven grouping distribution so that no node exists in some of the groups. As a result, the packet traversal speed cannot be estimated properly and thus the end-to-end real-time performance will be degraded. In addition, a larger GRA can introduce more control overhead (more temporary group ID update) and longer grouping time (larger back-off window), according to the grouping operation described in Section 2.4.1.

In order to further investigate the relationship between network density and optimal



(a) On time delivery rate with varying GRA



(b) On time delivery rate with varying node degree (N_D)

Figure 2.10: The effect of varying grouping granularity on end-to-end on-time packet delivery rate for single traffic flow with different end-to-end deadline requirements as 30ms and 60ms.

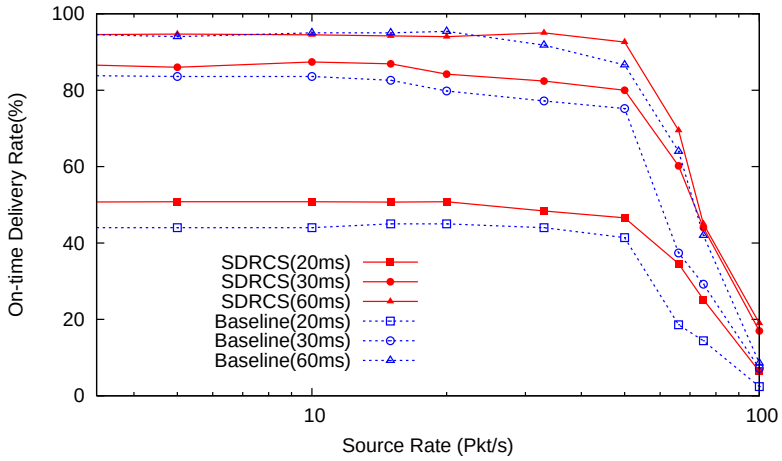
GRA value, we re-conduct the simulation with varying node degree values as 15, 22 and 30, and show the average on-time delivery rate of $CBR1$ and $CBR2$. The simulation results confirm that a larger GRA value can achieve better localization information for higher network density. With node degree as 15, 22 and 30, the best GRA values are found around 2, 3.5 and 5. Note that GRA may not be an integer based on its definition. For the following simulation scenarios, we keep the node degree as 15 and use $GRA = 2$, to achieve good end-to-end hop-distance estimation for SDRCS operation.

2.5.2 Effects of Early-Deadline-Miss Drop Policy

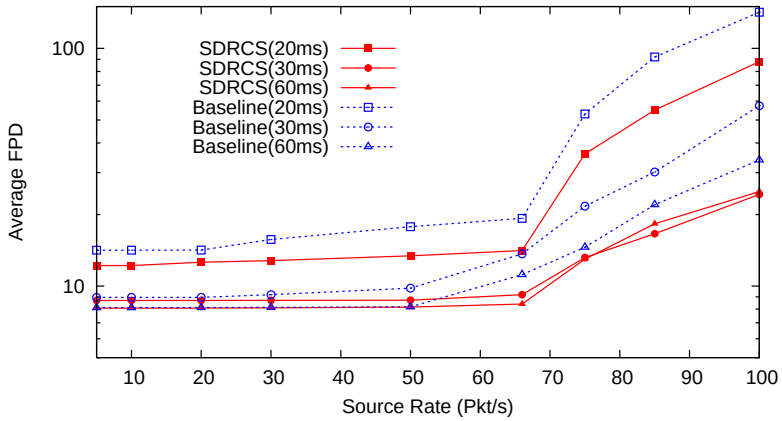
In this simulation scenario, we explore how SDRCS early deadline drop policy affects the end-to-end real-time performance in terms of on-time delivery rate. We conduct the simulation for three rounds. In each round, a CBR flow is generated at a node located at the bottom-left corner, with end-to-end deadline requirements as 20ms, 30ms and 60ms respectively. According to (2.7), the end-to-end deadline requirements in these rounds are mapped onto priority levels 1, 2 and 4. We compare the average packet on-time delivery rate achieved by using the proposed EDM drop policy and using a simple baseline end-to-end deadline miss drop policy in SDRCS operation and show the simulation results in Fig. 2.11 (a). The baseline end-to-end deadline miss drop policy is used in RAP [26], where a packet is dropped only when the deadline is past at an intermediate node. This baseline policy provides a naive soft real-time guarantee without using any local knowledge of packet average traversal speed for schedulability prediction.

From the simulation results shown in Fig. 2.11, we can observe that, compared with the baseline deadline miss drop policy, the proposed EDM drop policy in SDRCS design can improve the on-time delivery rate for up to 10 percent, especially for the traffic flows with high data rate and tight deadline requirements. Besides the admission control adopted at the sender, the instant deadline requirement is updated at each hop of packet forwarding, and compared with achievable deadline requirement according to (2.16) and (2.17). A packet that experiences large delay enroute and is unlikely to be delivered before the deadline can be dropped in the early stage of end-to-end transmissions. Therefore, the limited bandwidth can be utilized more efficiently.

EDM drop policy also plays an important role in improving the energy efficiency. Fig. 2.11 (b) shows the average number of data packet Forwarding Per end-to-end packet Delivery (*FPD* for short) by using these two drop policies. FPD reflects the performance of end-to-end packet delivery in terms of average end-to-end hop count, average retransmis-



(a) On time delivery rate w/o EDM drop policy



(b) Average PFD

Figure 2.11: The effect of early-deadline-miss (EDM) drop policy on end-to-end real-time performance for single traffic flow with different end-to-end deadline requirements as 20ms, 30ms, and 60ms.

sion times and percentage of packets dropped due to deadline miss. According to our admission control policy, while the data rate is low and the end-to-end deadline requirement is loose, most packets should be able to reach the sink on-time. Therefore, FPD approximates the product of average end-to-end hop count and average per-hop retransmission times in this case. While the data rate is high and the end-to-end deadline requirement is tight, the network congestion level is increased, resulting in longer transmission delay. Accordingly, the percentage of unschedulable packets will be dramatically increased, resulting in a large

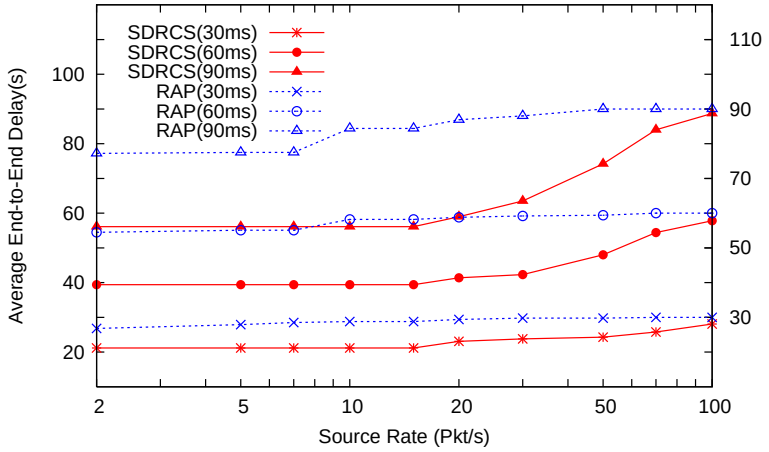
amount of data forwarding capacity wasted on the delivery of these packets. A properly designed drop policy can efficiently mitigate the number of unnecessary packet forwarding in this case. From Fig. 2.11 (b), we conclude that EDM drop policy always performs better than the baseline drop policy in terms of eliminating the unschedulable packet from the transmission, thus improving the overall communication energy efficiency.

2.5.3 Performance Comparison: SDRCS vs. RAP, MMSPEED

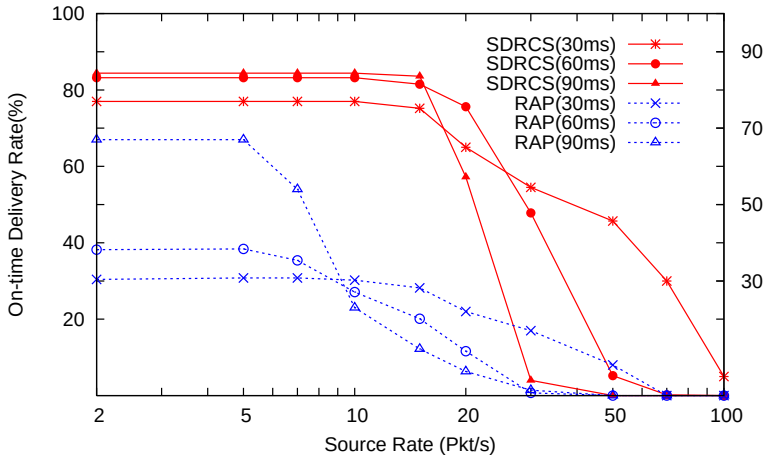
In this simulation scenario, we compare the real-time performance of SDRCS with existing service-differentiated real-time communication schemes RAP [26] and MMSpeed [21]. In order to fully test the two protocols, we randomly generate 10 network topologies with node degree as 15. For each network topology, we chose three nodes located in the lower-right corner as the event data sources for maximizing the possible end-to-end hop count.

Three CBR (constant bit rate) event data flows CBR1, CBR2, and CBR3 with different end-to-end deadline requirements are generated simultaneously at the source nodes. CBR1 is assigned a tight end-to-end deadline requirement of 30ms. CBR2 is assigned a medium end-to-end deadline requirement of 60ms. CBR3 is assigned a loose end-to-end deadline requirement of 90ms. According to the sample packet priority assignment policy given in (2.7), $P^{Tx}_{CBR1} = 2$ $P^{Tx}_{CBR2} = 4$, and $P^{Tx}_{CBR3} = 5$. Note that, the P^{Tx} values listed here are the initial packet priority level assignment at the sender. The priority level of a packet is updated at each hop, according to (2.17) and (2.7). For each CBR flow, 2000 packets are generated and sent to the sink.

Fig. 2.12 shows the average end-to-end transmission latency and average on-time delivery rate collected from the simulation by using RAP and SDRCS. From Fig. 2.12 (a), we can observe that both RAP and SDRCS can provide service-differentiation for the traffic flows with different end-to-end deadline requirement in terms of different average end-to-end transmission latencies, because both designs provide prioritized queuing and MAC



(a) End-to-end latency (RAP v.s. SDRCS).



(b) On-time delivery rate (RAP v.s. SDRCS).

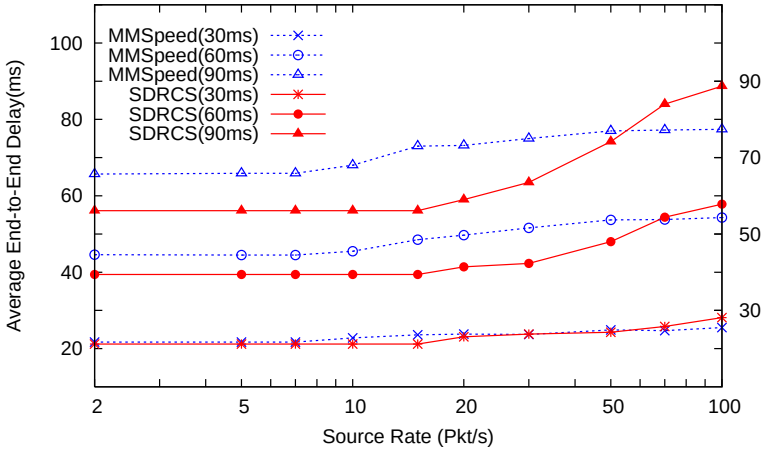
Figure 2.12: End-to-end real-time performance comparison between RAP and SDRCS.

support. However, RAP always results in a higher end-to-end latency. In our simulation environment setting, a log-normal shadow channel model is used to reflect the channel dynamics in real WSN deployments. The channel quality enroute affects the end-to-end delay in terms of per-hop retransmission. The tradeoff between larger per-hop forwarding distance and shorter per-hop latency plays an important role in determining the end-to-end real-time performance. Since RAP assumes a perfect channel model, it simply chooses the next-hop to maximize the per-hop forwarding distance but fails to consider this tradeoff in its forwarding metric design. As a result, the average end-to-end delay is dramatically

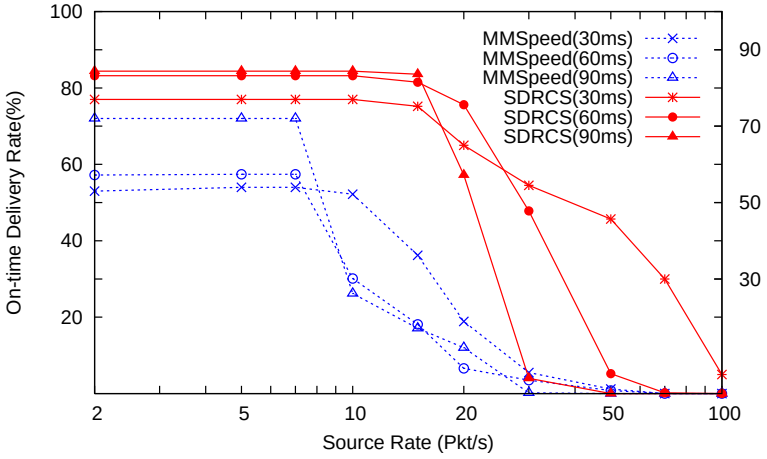
increased for RAP. Moreover, since RAP does not have a dynamic packet traversal speed estimation strategy in the protocol design, only a baseline deadline-miss packet drop policy is used in end-to-end transmission. Therefore, the unnecessary packet forwarding cannot be eliminated to improve the bandwidth utilization. The long transmission latency and the low bandwidth utilization also affect the RAP performance in terms of on-time delivery rate. From Fig. 2.12 (b), SDRCS provides up to 40% higher average on-time delivery rate for 2000 packet transmissions and maintains steady on-time delivery rates without congestion for much larger source rates.

Regarding comparison with MMSpeed, from Fig. 2.13 (a), it can be observed that both schemes can provide a relatively low average end-to-end latency for achieving soft real-time guarantees. This result shows that, in contrast to pure geographic forwarding, a cross-optimization on both geographic information and channel quality in dynamic forwarding design for delay sensitive end-to-end communication is necessary. From Fig. 2.13 (b), it can be observed that, compared with MMSpeed, SDRCS improves the steady on-time delivery rate of any priority level traffic for about 20 percent when the event source rate is less than $5Pkt/s$. When the event source rate reaches $5Pkt/s$, MMSpeed starts to experience network congestion with dramatically decreased on-time delivery rate. In contrast, SDRCS maintains the steady on-time delivery rate for larger event source rates up to $20Pkt/s$.

The better real-time performance of SDRCS can be attributed to the following reasons. First, *SDRCS provides better overall per-hop transmission latency for traffic with low priority level.* According to the MAC operations described in Section 2.4.3, the average per-hop latency for SDRCS is $2200\mu s$ regardless of the priority levels associated with the traffic. According to (2.8), this value is about the same as the minimum per-hop latency for dynamic IFS/BW extension based approach with priority level equal to 2. However, for MMSpeed, the traffic with lower priority level has to experiences longer IFS and and back-off time in each packet transmission attempt, which results in significantly in-



(a) End-to-end latency (MMSpeed v.s. SDRCS).



(b) On-time delivery rate (MMSpeed v.s. SDRCS).

Figure 2.13: End-to-end real-time performance comparison between MMSpeed and SDRCS.

creased transmission delay. Therefore, the dynamic IFS/BW extension based MAC design adopted by MMSpeed leads to lower transmission throughput and on-time delivery rate, especially for low priority-level traffic. Accordingly, as shown in Fig. 2.13 (b), the traffic with looser end-to-end deadline requirements experience congestion earlier in MMSpeed; while in SDRCS, all priority levels of traffic enters the congestion status at about the same source rate. In addition, SDRCS achieves better on-time delivery rate at the same packet source rate.

Second, *SDRCS transmits less duplicated packets.* MMSpeed design uses probabilistic

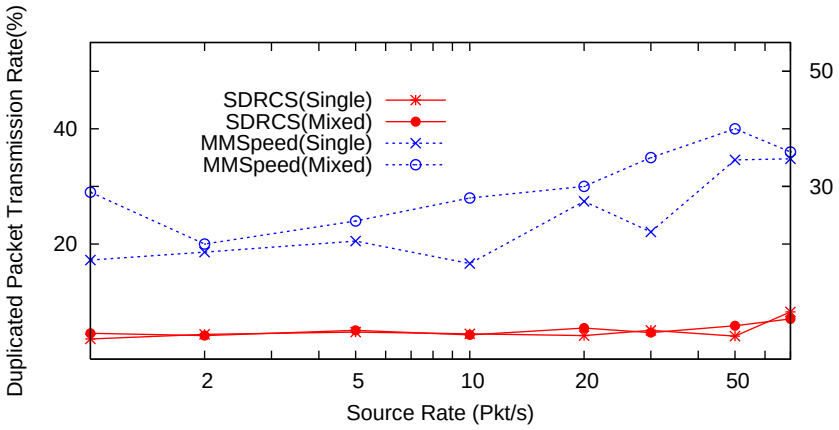


Figure 2.14: Performance comparison between SDRCS and MMSpeed in terms of percentage of duplicated packets received at the sink in all on-time delivered packets with single or mixed priority CBR traffic flows.

per-hop multicast to improve the end-to-end on-time delivery rate in soft real-time applications. The number of multicast receivers in each hop will be determined by an end-to-end link error rate estimation. However, due to the channel dynamics in WSNs, such link error rate estimation is not accurate. The improper per-hop multicast could result in duplicated packets being delivered to the sink, thus greatly deteriorating the bandwidth utilization. In order to verify this assumption, we construct a simulation scenario to evaluate the percentage of duplicated packets in all on-time delivered packets, shown in Fig. 2.14. Two kinds of traffic flows are set up in this simulation scenario: a single-flow sent from one lower-right node with end-to-end deadline as 30ms, and a mixed-flow sent from two lower-right nodes with end-to-end deadline as 30ms and 60ms separately. The simulation results verify that SDRCS reduces more than 70% duplicated packet transmissions in network, thus gaining much better bandwidth utilization with higher end-to-end on-time delivery rate. The main source of duplicated packet transmission in SDRCS design lies in the retransmission introduced by lost acknowledgement.

Third, *SDRCS uses better deadline miss drop policy*. MMSpeed use table-based forwarding technique, which is motivated by the earlier real-time communication scheme

SPEED. The neighbor list and the average transmission time between a pair of neighboring nodes are periodically exchanged using broadcast. The packet traversal speed achieved by a certain neighbor is determined by the sender based on this periodically exchanged information. For an end-to-end packet delivery process, if any intermediate node cannot find a neighbor to satisfy the required packet traversal speed, the packet is dropped. Such rigid packet-drop policy adopted by MMSpeed requires accurate end-to-end hop count estimation and frequent neighbor information exchange. Under dynamic network conditions, the drop policy adopted by MMSpeed may experience large percentage of false deadline miss packet drop. While for SDRCS, both the packet traversal speed and the required per-hop deadline are estimated based on the information instantaneously updated at the receiver, which helps improve the accuracy of schedulability estimation. From Fig. 2.13 (a), we can observe that, for SDRCS, the average end-to-end latency increases with increasing source rate until it approaches the end-to-end deadline requirement. The reason behind this lie in the fact that most packet drops according to EDM drop policy is correctly estimated. Therefore, the average end-to-end transmission latency for on-time delivered packet increases with larger network queuing delay. However, for MMSpeed, the average end-to-end latency does not increase properly with increasing source rate, even when the network is fully congested. This situation shows that the drop policy adopted by MMSpeed is too tight so that a large portion of the schedulable packets with end-to-end transmission latency close to the deadline requirement are dropped enroute.

Last, *SDRCS design results in much less control overhead than MMSpeed*: Compared with the neighbor list based forwarding technique in MMSpeed, the dynamic forwarding technique adopted by SDRCS provides stateless routing solution so that the huge amount of broadcast message for periodic neighbor information exchange can be eliminated. In high density WSNs, where the number of neighboring nodes is large, the resulting bandwidth utilization improvement by dynamic forwarding can be huge compared with the table-based

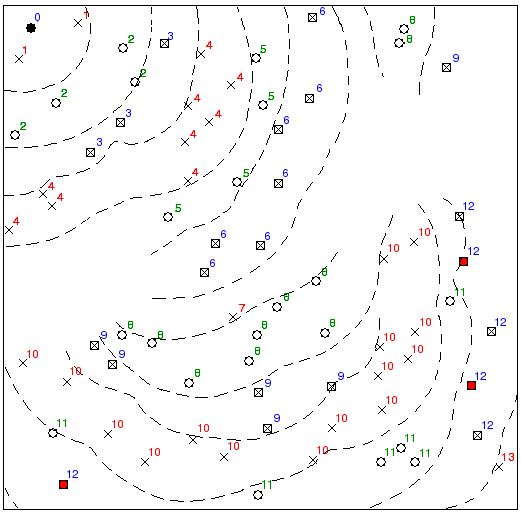
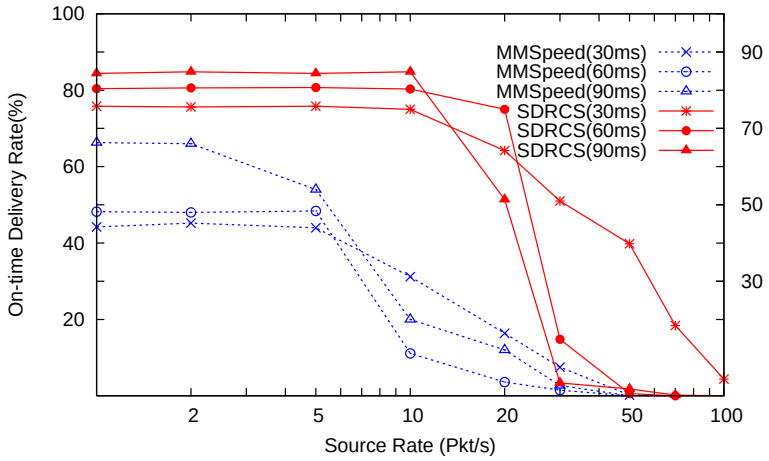


Figure 2.15: A sample network topology with two communication voids and the RSS-based group formation results with $GRA = 2$. The source nodes are marked in solid square.

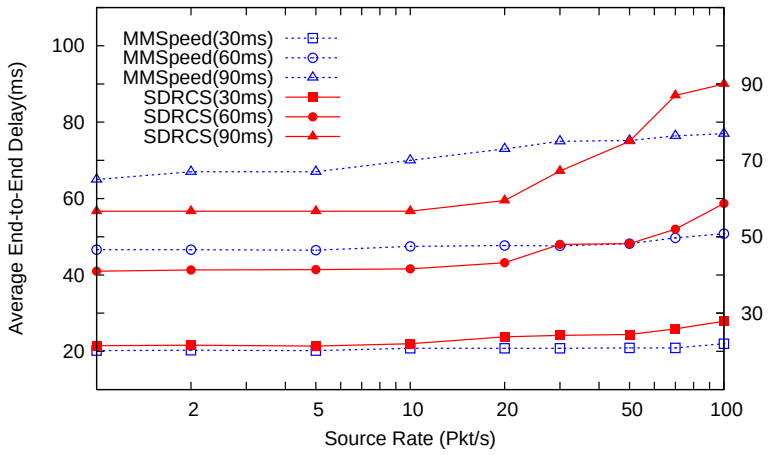
forwarding approaches.

2.5.4 Void Avoidance Capability

In this simulation scenario, we investigate the void avoidance capability of SDRCS design and compare its end-to-end real-time performance with MMSpeed in the presence of the communication voids. We manually generate a network topology by taking out 30 nodes from the network as shown in Fig. 2.8 (a), to create two communication voids located at the lower-left and top-right corner. Fig. 2.15 shows the RSS-based group formation results for this topology. As described in Section 2.4.6, each node is able to get a group ID assignment in the sample network topology, since the network remains connected. The resulting group ID contour bends around the farther end of the communication voids from the sink, which is no longer linearly related to the node-to-sink distance. A route always exists from the node to the sink following the vertical direction of the group ID contour around the void area. The group ID assignment adapts to the network void and reflects the end-to-end hop count estimation instead of distance estimation.



(a) Average on-time delivery rate



(b) Average end-to-end transmission latency

Figure 2.16: End-to-end real-time performance of SDRCS and MMSpeed based on the sample network topology with communication voids.

We repeat the simulation scenario described in Section 2.5.3 using three source nodes marked in solid square in Fig. 2.15 and observe the average end-to-end transmission latency and on-time delivery rate for both SDRCS and MMSpeed. The simulation results are shown in Fig. 2.16. Compared with the real-time performance based on non-void topology, as shown in Fig. 2.13, SDRCS does not suffer from a severe service degradation. Two main reasons contribute to the results. First, SDRCS design guarantees a route from the sender to the sink no matter which route is taken by the dynamic forwarding process. Therefore, a

communication void can only slightly increase the end-to-end hop count according to the grouping results. Second, the network capacity bottleneck is located around the sink, where all the packets are converge-cast to this area. Therefore, even though less parallel forwarding paths can be taken from the sender to the sink, the end-to-end throughput cannot be affected dramatically. However, for MMSpeed, the on-time delivery rates drop by about 10 percent for all priority levels of traffic flows. Since MMSpeed relies on negative participation for void avoidance, some of the forwarding decisions may lead to broken routes. As a result, the relaying node will drop the packets and stop participating in further transmissions. Since it takes time for the negative participation to propagate in the reverse direction to the upstream nodes and trigger the new route exploration, certain percentage of packets are dropped due to communication void. This situation becomes extremely severe when the total number of packet transmissions is low. Also note that we do not setup a scenario for MMSpeed where the source nodes cannot find a route at all by negative participation.

2.6 Summary

In this chapter, we propose a novel cross-layer communication scheme, SDRCS, to provide service differentiated soft real-time guarantees for multi-hop communication in WSNs. By using RSS-based grouping, we enable end-to-end hop-distance awareness for sensor nodes with low control overhead. The hop-distance estimation accuracy can be regulated by the grouping granularity parameter to meet various application requirements. With the grouping knowledge, a channel quality aware dynamic forwarding approach is proposed with a polling contention-period based prioritized MAC for inter-node traffic differentiation. Compared with the commonly adopted dynamic IFS/BW extension based MAC approaches, the proposed MAC design features better service differentiation capability with better bandwidth utilization when the number of priority levels in the network is great

than 4. By combining a receiver contention process in the MAC operation, the forwarding decision is locally optimized for packet traversal speed maximization. According to the proposed MAC operation, we also design a per-hop deadline based prioritized queueing policy for intra-node traffic differentiation.

The proposed SDRCS design requires minimum hardware support at the sensor nodes, where no localization, transmission power adaptation or multiple channel transmission support is required. It also adapts well to network dynamics, such as channel quality, local congestion and communication voids. According to our design and performance analysis, compared with existing service differentiated real-time communication schemes RAP [26] and MMSpeed [21], SDRCS is able to achieve better on-time delivery ratio with better energy efficiency for mixed priority traffic flows in unsynchronized WSNs. SDRCS also provides higher end-to-end throughput in terms of supporting higher source data rates without congestion. Our future work includes implementing the design in a sensor network testbed with duty-cycle design and making use of the data generated by real event detection applications for further performance evaluation and improvement.

Chapter 3

Energy/Latency Efficiency of Anycast-Based Forwarding over Lossy Channel

3.1 Introduction

Anycast-based forwarding (*anycasting* for short) has been proposed as an efficient communication technique for asynchronous duty-cycled Wireless Sensor Networks (WSNs) [51, 17]. Compared with the traditional unicast-based forwarding schemes, where each sensor node has a single next-hop node specified by the routing metric and it has to wait for the designated next-hop node to wake up before the data packet transmission, anycasting technique does not specify a fixed next-hop node at the sender. Instead, the sender utilizes the broadcast nature of preamble/RTS packet transmission to wake up neighboring nodes and forms a forwarding set. All awake nodes within the forwarding set are then prioritized according to a specific forwarding metric. The node with the highest priority becomes the next-hop node to respond with a CTS packet and receive the data packet. According to

the basic anycasting operation, the performance of an anycasting design highly depends on two important design factors:

(1) *Prioritization Policy*: How the neighboring nodes of a sender should be prioritized so that the most suitable awake node in the forwarding set can be selected as the next-hop to reduce the end-to-end communication cost.

(2) *Wake-up Policy*: How much time and energy should be spent in the neighboring node wake-up process so that the most suitable node can be selected from a reasonably sized forwarding set to forward the packet and reduce the overall communication cost?

In this chapter, we provide systematic investigation of these two important questions by presenting a statistical end-to-end cost model for basic anycasting operation under a realistic wireless channel model. Based on the model, we use the cumulative distribution function of two proposed end-to-end metrics: *latency-coefficient* and *energy-coefficient*, to characterize the cost-efficiency of an anycasting design with specific design parameters. By exploring the relationship between the end-to-end cost efficiency and the forwarding decision dependent anycasting design parameters, two greedy forwarding metrics are proposed for cost-efficient neighboring node prioritization in a fully distributed manner. By exploring the relationship among the preamble length, the size of the forwarding set, and the achievable end-to-end cost efficiency, a series of preamble length control guidelines are developed for cost-efficient anycasting wake-up policy in WSNs with different duty-cycle operations. According to our analytical results and simulation validation, the proposed forwarding metrics and preamble length control guidelines significantly reduce the end-to-end energy and latency cost in duty-cycled WSNs.

The rest of the chapter is organized as follows. Section 3.2 describes the existing anycasting analysis frameworks and points out their limitations. Section 3.3 describes the assumptions and the network model considered in our work. Section 3.4 describes an end-to-end anycasting cost-model for single flow transmission under a realistic log-normal

channel model. Sections 3.5 and 3.6 discuss how to design cost-efficient neighboring node prioritization and wake-up policies,respectively, based on our proposed cost model. The analytical results derived in Sections 3.5 and 3.6 are validated through the simulation results presented in Section 3.7. The chapter is concluded in Section 3.8.

3.2 Related Work

GeRaF [33] is one of the earliest anycasting designs to utilize the broadcast nature of the RTS packet transmission and enable multiple neighboring nodes to compete for being the most suitable forwarder so that the end-to-end communication throughput can be improved. A series of studies [51, 52, 53, 54, 55] have been conducted to model and optimize the design parameters of GeRaF-type anycasting, sometime classified as opportunistic routing, designs. These studies prioritize the neighboring nodes by their geographic locations and try to minimize the end-to-end hop count through global optimized or distributed greedy forwarding metrics and wake-up policy. Based on an ideal disc channel model, end-to-end hop count minimization guarantees optimal end-to-end performance because possible retransmissions are not considered for the DATA package forwarding between the sender and selected forwarder. However, wireless links in practical WSN settings can be extremely unreliable, deviating to a large extent from the ideal channel models used in these works.

The work proposed in [56] provides a geographic forwarding performance analysis under lossy channel and suggests using new channel-quality-aware forwarding metrics to prioritize neighboring nodes. However, the effect of duty-cycle operation and wake-up policy on the end-to-end communication cost are not considered in the analysis framework. Traditionally, the unicast forwarding sender, such as in BMAC [57], uses a long preamble to wake up the neighboring nodes so that the packet can always be forwarded to the most suitable node among all of its neighbors for communication cost minimization. In contrast,

for most existing anycasting schemes, the sender uses a short RTS packet to wake up its neighbors and form a non-empty forwarding set, in which the most suitable node among all neighboring nodes may not be included. As a result, the packet may end up being forwarded to a sub-optimal next-hop node but cost less for forming a smaller forwarding set. Clearly, a systematic analysis on how such a tradeoff affects the end-to-end cost-efficiency under realistic channel model is vital for efficient anycast design, which however, has not been done in previous work.

3.3 Assumptions and System Model

3.3.1 Assumptions

We consider a WSN composed of sensor nodes with location awareness but not global topology awareness. The nodes are randomly deployed with a uniform distribution. Duty-cycle operation is adopted in the network so that each sensor node switches between sleep and awake states periodically. The sensing data are collected by the sensor nodes and converge-casted to the sink. This study focuses on low-rate/time-scheduled applications such as habitat monitoring, where interference is at minimum (or nonexistent) [56]. Interference is an important factor to consider, specially in medium and heavy traffic scenarios, and is a subject of our future work.

3.3.2 Basic Anycasting Operation

The basic anycasting operation consist of two continuous handshake processes, as shown in Fig. 3.1.

Control Packet Handshake: The anycasting operation is initiated by a control packet handshake process. The sender (S in Fig. 3.1) sends a preamble followed by an RTS packet

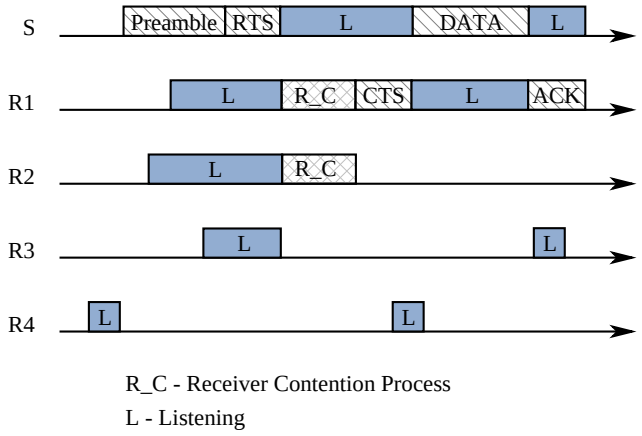


Figure 3.1: Timeline of RTS-CTS and DATA-ACK packet exchange between a sender, S, and its neighboring nodes, R1-R4, in anycasting.

to wake up the neighboring nodes. The information required for prioritizing the awake neighboring nodes is contained in the RTS packet. The sender also sets up a timer for the handshake process. If the timer expires before a CTS packet is received, the sender retransmits the RTS packet and resets the timer. Any neighboring node that overhears the preamble frame stays awake and is defined as a *Active Receiver* (AR). The ARs that successfully receive the RTS packet and are located within the forwarding region of the sender, as shown in Fig. 3.2, become *Potential Receiver*s (PR). The rest of the ARs switch to sleep mode immediately and follow the default duty-cycle from then on (R3 in Fig. 3.1). All the potential receivers then start the *receiver contention process*, where each potential receiver prioritizes its potential as the next hop according to a specific forwarding metric. By using a polling contention period based channel contention strategy proposed in [58], a PR that has the highest priority can obtain the channel within a guaranteed number of time slots and respond to the sender with a CTS packet containing its node ID (R1 in Fig. 3.1). We call this node as the *Winning Receiver* (WR). Other potential receivers will switch back to the sleep period and follow the default duty-cycle design (R2 in Fig. 3.1).

Data Packet Handshake: After a successful control packet handshake between the

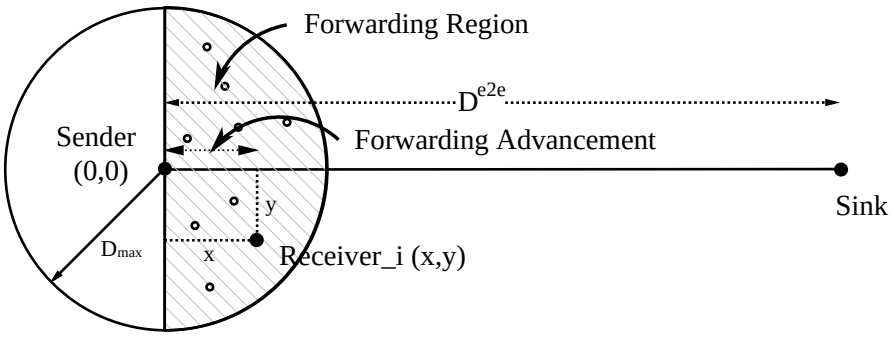


Figure 3.2: Coordination illustration for single-hop anycasting operation.

sender and the winning receiver, the data packet forwarding is accomplished by the data packet handshake. In a data packet handshake, the sender will unicast the DATA packet to the winning receiver, whose node ID is indicated in the CTS packet. The sender also sets up a timer for the data packet handshake process. Upon receiving the DATA packet, the winning receiver responds to the sender with an ACK packet. If the timer expires before an ACK packet is received, the sender retransmits the DATA packet and resets the timer.

Note that, some variations of anycasting design, such as those proposed in [59, 60], consider forwarding without Preamble and RTS/CTS exchange in MAC layer operation. All neighboring nodes, if any, that receive the DATA packet compete for being the most suitable forwarder. Since the length of the DATA packet is usually much longer than the RTS/CTS packet, the data packet error rate is usually high at the neighboring nodes. In duty-cycled WSNs, the size of forwarding set thus becomes very small or close to zero and the number of retransmission becomes large. Therefore, different analytical models and design policies need to be developed to these variations based on different forwarding operations, which are out of the scope of our work.

3.3.3 Channel Model

The log-normal channel model proposed in [12] is used to capture the signal attenuation and dynamics of the wireless channel. We consider the DSSS-OQPSK (Offset Quadrature Phase Shift Keying with Direct Sequence Spread Spectrum) as the modulation scheme, which is adopted by the commonly used CC2420 transceiver in MicaZ and TelosB motes [61]. The PRR achieved at distance D, for a packet with length l bits can be found as

$$\Psi(D,l) \;\; = \left(1 - Q\left(\sqrt{\tfrac{2B}{R} \cdot 10^{\frac{\Gamma_\sigma(D)}{10}}}\right)\right)^l. \tag{3.1}$$

where

$$\begin{aligned} \Gamma_\sigma(D) &= P_t - P_L(D_0) - 10\eta \log_{10}(\frac{D}{D_0}) - P_n + X_\sigma, \\ Q(D) &= \frac{1}{\sqrt{2\pi}} \int_D^\infty e^{-\frac{t^2}{2}} dt. \end{aligned} \tag{3.2}$$

In the above formulae; R is the radio communication bit rate; B is the noise bandwidth; P_t is the transmit power in dBm; $P_L(D_0)$ is the path loss at a reference distance D_0 in dBm; η is the path loss exponent; D is the distance between the wireless sender and receiver; P_n is the receiver-end noise power; and $X_\sigma \sim \mathcal{N}(0,\sigma)$ is the shadow fading component represented as a Normal-distribution random variable with mean 0 and variance σ. The expectation of $\Psi(D,l)$ is thus given as

$$\overline{\Psi}(D,l) = \frac{1}{2\pi\sigma} \int_{-\infty}^{+\infty} \Psi(D,l) * e^{\frac{-(\gamma-\mu(D))^2}{2\sigma^2}} d\gamma, \tag{3.3}$$

where

$$\mu(D) = P_t - P_L(D_0) - 10\eta \log_{10}(\frac{D}{D_0}) - P_n. \tag{3.4}$$

Compared with the ideal disc channel model used in [33] [54] [55] [51], this channel model captures the channel dynamics in real WSN deployment, where the sender and the receiver communicate over unreliable links.

Note that $\Psi(D, l)$ is a random variable. Therefore, no cut-off value exists as the maximum transmission for a sender. Instead, a valid definition for disconnection distance, D_{max}, under log-normal channel model is given as the maximum transmission range, at which the wireless links have a high probability ($> p_H$) of having low packet reception rates ($< \Psi_L$) [12]. By defining proper threshold values $p_H = 0.96$, and $\Psi_L = 0.1$ [56], D_{max} can be derived as

$$D_{max} = 10^{\frac{\gamma_L - \sigma Q^{-1}(1-p_H) - P_t + P_n + P_L(D_0)}{-10\eta}}. \tag{3.5}$$

where γ_ℓ is the signal-to-noise ratio in dBm corresponding to Ψ_L; and $Q(.)$ is given in (3.2). Based on (3.5), we also define the network node density in terms of **node degree** ρ, where ρ is the expected number of nodes within the area of maximum transmission range, D_{max}.

3.3.4 Asynchronous Duty-Cycle Operation and Forwarding Set

Asynchronous duty-cycle operation is adopted by each sensor node with a t_L awake-period followed by a t_S sleep-period. The entire period of a duty-cycle is defined as t_D, where $t_D = t_L + t_S$. The duty-cycle is then derived as $d_c = \frac{t_L}{t_D}$. In order to ensure the recognition of a preamble during the listening period, the receiver must start its listening period before the last preamble slot is transmitted, as shown in Fig. 3.2. We denote this minimum intersection of listening period and preamble period as t_C.

Based on the aforementioned duty-cycle operation, we then derive the probability that

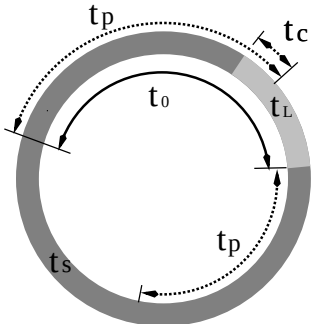


Figure 3.3: Deriving P_{wake} through sliding t_P in t_D period.

a neighboring node is awakened by a preamble of length t_P as P_{wake}, as shown in Fig. 3.3,

$$P_{wake} = \begin{cases} \frac{(t_P + t_D d_c - t_C)}{t_D} & t_C \leq t_P \leq t_M, \\ 1 & t_P > t_M, \end{cases} \tag{3.6}$$

where $t_M = t_C + t_D(1 - d_c)$. From Eqn. (3.6), we observe that when $t_P > t_M$, all the neighboring nodes within the forwarding region are guaranteed to be awakened by the preamble. Since P_{wake} can be varied only when $t_C \leq t_P \leq t_M$, we define the normalized preamble length, $t_{P,n}$, as

$$t_{P,n} = \frac{t_P - t_C}{t_M - t_C} \tag{3.7}$$

Based on Eqn. (3.6), we derive some important values related to the size of forwarding set of anycasting operations. First, we define ρ_I as the expected number of neighboring nodes awakened by the preamble, i.e. the expected number of ARs, which is given as

$$\rho_I = \rho \cdot P_{wake} \tag{3.8}$$

We then define ρ_I^R as the expected number of PRs, i.e., the expected number of nodes

participating in the receiver contention process. From the perspective of an anycasting sender, the PRs form the forwarding set, from which an optimal receiver is selected by the prioritization policy specified by a forwarding metric. To find ρ_I^R, we first consider the probability of a node, that is randomly placed within the maximum transmission range of the anycasting sender, successfully receiving the RTS packet. Such probability is given as

$$P_{R0} = \frac{1}{2\pi\sigma} \int_{-\infty}^{\infty} \int_{0}^{D_{max}} \Psi(D, l_{RTS}) p(D) e^{\frac{-(\gamma-\mu(D))^2}{2\sigma^2}} dD d\gamma, \tag{3.9}$$

where $p(D) = 2D/D_{max}^2$ is the probability of the node located at distance D from the sender. Then ρ_I^R is given as

$$\rho_I^R = \rho' \cdot P_{wake} \cdot P_{R0}, \tag{3.10}$$

where $\rho' = \frac{\rho}{2}$ is the expected number of neighbor nodes located within the forwarding region.

3.4 End-to-end Cost Model

In this section, we first provide a statistical analysis of transmission latency and energy consumption for single-hop anycasting operation. Based on that, we define cost-coefficient as the end-to-end metric to determine the latency efficiency and energy efficiency of an anycasting design with specific protocol parameters. The proposed end-to-end cost model is then used to facilitate the cost-efficient anycasting prioritization policy and wake-up policy design in duty-cycled WSNs.

3.4.1 Single-Hop Transmission Latency

First, we derive the expected transmission latency for a successful single-hop anycasting. For the control packet handshake, an RTS retransmission is issued whenever the sender is timed out for receiving the CTS packet. Therefore, the expected time for a successful control packet handshake, $E[\mathcal{T}_{h/s}^{ctrl}]$, is given by the required number of RTS packet retransmissions for a successful handshake and the time consumed for each handshake attempt. Let $T^{ctrl} = T_{P|R} + T_{R\_C} + T_C$, where $T_{P|R}$ is the time for transmitting the preamble and the RTS packet, T_{R_C} is the time required for receiver contention, T_C is the time for transmitting the CTS packet, and $T_{t/o}^{ctrl}$ is the timeout value at the sender for RTS packet retransmission, then

$$\begin{aligned} E[\mathcal{T}_{h/s}^{ctrl}] &= \sum_{n=1}^{\infty} (1 - P_R \cdot P_C)^{n-1} (P_R \cdot P_C) \left[(n-1)(T_{P|R} + T_{t/o}^{ctrl}) + T^{ctrl} \right] \\ &= \left(\frac{1}{P_R \cdot P_C} - 1 \right) \cdot (T_{P|R} + T_{t/o}^{ctrl}) + T^{ctrl} \end{aligned}$$

where P_R is the probability that the RTS packet is successfully received by at least one potential receiver, and P_C is the probability that the CTS packet is successfully received by the sender. With the protocol-specific RTS and CTS packet lengths, P_R and P_C achieved by a potential receiver are calculated according to (3.3). If we assume $T_{t/o}^{ctrl} = T_{R\_C} + T_C$, i.e., the sender retransmits the (preamble+RTS) frame if no CTS is received after $T_{R\_C} + T_C$ time period, $E[\mathcal{T}_{h/s}^{ctrl}]$ is then simplified as

$$E[\mathcal{T}_{h/s}^{ctrl}] = \frac{T^{ctrl}}{P_R \cdot P_C}. \tag{3.11}$$

The expected time for a successful data packet handshake is calculated similar to the control packet handshake as

$$E[\mathcal{T}_{h/s}^{data}] = \left(\frac{1}{P_D \cdot P_A} - 1\right) \cdot (T_D + T_{t/o}^{data}) + (T_D + T_A)$$

where P_D is the probability of the DATA packet being successfully received by the winning receiver, and P_A is the probability of the ACK packet being successfully received by the sender. With the protocol-specific DATA and ACK packet lengths, P_D and P_A achieved by a potential receiver can also be calculated according to (3.3). T_D is the time for transmitting the DATA packet, T_A is the time for transmitting the ACK packet, and $T_{t/o}^{data}$ is the timeout value at the sender for DATA packet retransmission. Similarly, if we assume $T_{t/o}^{data} = T_D + T_A$,

$$E[\mathcal{T}_{h/s}^{data}] = \frac{T^{data}}{P_D \cdot P_A}, \tag{3.12}$$

where $T^{data} = T_D + T_A$. According to (3.11) and (3.12), the expected time for a successful single-hop forwarding is

$$E[\mathcal{T}^{hop}] = E[\mathcal{T}_{h/s}^{ctrl}] + E[\mathcal{T}_{h/s}^{data}] = \frac{T^{ctrl}}{P_R \cdot P_C} + \frac{T^{data}}{P_D \cdot P_A}. \tag{3.13}$$

3.4.2 Single-Hop Energy Consumption

In this subsection, an energy consumption analysis is provided for a successful single-hop anycasting operation. The energy consumption at a node is derived using the product of its awake time, t_w, and the unit-time energy (i.e. power) consumption parameter ε. The energy consumption is not distinguished between transmitting and receiving status of a node while it is awake, because the unit-time energy consumption parameters, ε, in these two status are almost the same for popular transceivers ($\varepsilon \doteq 25mJ/s$ when $P_t = 0dBm$) used on most sensor nodes [61, 40]. The energy consumption at the sender and the receivers are

derived separately. For an anycasting sender, it remains awake during the entire forwarding process. According to (3.13), the energy consumed at the sender for a successful single-hop forwarding is given as

$$E[\mathcal{E}_s^{hop}] = E[\mathcal{T}^{hop}] \cdot \varepsilon \tag{3.14}$$

To calculate the energy consumption at the receivers, we first consider the control packet handshake process, where different amount of energy is consumed by the active receivers, the potential receivers and the winning receivers. Let $E^{ctrl} = E_{P|R} + E_{R\_C} + E_C$, where $E_{P|R} = \rho_I \cdot [\frac{1}{2} \cdot T_{P|R} \cdot \varepsilon]$, according to (3.8), is the total amount of energy consumed by the active receivers for receiving the preamble followed by the RTS packet; $E_{R\_C} = \rho_I^R \cdot [T_{R\_C} \cdot \varepsilon]$, according to (3.10), is the energy consumed by the potential receivers during the receiver contention process; $E_C = T_C \cdot \varepsilon$ is the energy consumed by the winning receiver in transmitting the CTS packet; the energy consumed at the receivers for a successful control packet handshake is then given as

$$\begin{aligned} E[\mathcal{E}_{h/s,r}^{ctrl}] = \sum_{n=1}^{\infty} \sum_{m=0}^{n-1} \Bigg[\binom{n-1}{m} \left[m E_{P|R} + (n-m) E^{ctrl} \right] \\ (1-P_R)^m [P_R(1-P_C)]^{n-m-1} (P_R P_C) \Bigg] \\ = \frac{E_{P|R} + P_R(E_{R\_C} + E_C)}{(P_R P_C)^2}. \end{aligned}$$

For the data packet handshake, since the winning receiver is the only awake neighboring node and remains awake during the handshake, the energy consumption at the receivers for a successful data packet handshake can be found as

$$E[\mathcal{E}_{h/s,r}^{data}] = E[\mathcal{T}_{h/s,r}^{data}] \cdot \varepsilon = \frac{T^{data}}{P_D \cdot P_A} \cdot \varepsilon.$$

Accordingly, the expected energy consumption at the receivers for a successful single-hop forwarding is given as

$$\begin{aligned} E[\mathcal{E}_r^{hop}] &= E[\mathcal{E}_{h/s,r}^{ctrl}] + E[\mathcal{E}_{h/s,r}^{data}] \\ &= \left[\frac{\frac{1}{2}\rho_I T_{P|R} + P_R(\rho_I^R T_{R\_C} + T_C)}{(P_R P_C)^2} + \frac{T^{data}}{P_D P_A} \right] \varepsilon. \end{aligned} \tag{3.15}$$

Finally, according to (3.14) and (3.15), the expected energy consumption in the network for a successful single-hop forwarding is

$$E[\mathcal{E}^{hop}] = E[\mathcal{E}_s^{hop}] + E[\mathcal{E}_r^{hop}] \tag{3.16}$$

3.4.3 End-to-end Latency and Energy Cost

In multi-hop WSNs, the end-to-end transmission cost depends not only on single-hop transmission cost but also on the number of hops required end-to-end. Therefore, the expected communication cost as a result of a single flow from a source node at distance D^{e2e} from the sink, $E[C^{e2e}]$, is given as

$$E[C^{e2e}] = E[C^{hop}] \cdot E[H^{e2e}] = D^{e2e} \cdot \frac{E[C^{hop}]}{E[A^{hop}]}, \tag{3.17}$$

where $E[H^{e2e}]$ is the expected end-to-end hop count; $E[C^{hop}]$ is the expected per-hop communication cost, such as per-hop transmission latency or energy consumption; and $E[A^{hop}]$ is the expected per-hop forwarding advancement, which is defined as the end-to-end distance decrement achieved in single-hop forwarding towards the sink, as shown in Fig. 3.2. For a single flow with fixed end-to-end distance, an efficient anycasting design aims at minimizing the component, $\frac{E[C^{hop}]}{E[A^{hop}]}$, i.e., the per-hop communication cost normalized by the expected per-hop forwarding advancement.

In order to capture the cost-efficiency of anycasting, based on (3.17), we define the end-to-end cost coefficient, $\mathcal{C}$, as

$$\mathcal{C} = \frac{E[C^{hop}]}{E[A^{hop}]}, \tag{3.18}$$

which is independent of the end-to-end transmission distance. By substituting $E[C^{hop}]$ with (3.13) and (3.16) separately, the end-to-end latency coefficient, $\mathcal{C}^T$, is found as

$$\mathcal{C}^T = \frac{1}{E[A^{hop}]} \cdot \left(\frac{T^{ctrl}}{P_R \cdot P_C} + \frac{T^{data}}{P_D \cdot P_A} \right) \tag{3.19}$$

and the end-to-end energy coefficient, $\mathcal{C}^E$, is found as

$$\mathcal{C}^E = \frac{\varepsilon \left[\frac{T^{ctrl}}{P_R P_C} + \frac{2T^{data}}{P_D P_A} + \frac{\frac{1}{2}\rho_I T_{P|R} + P_R(\rho_I^R T_{R\_C} + T_C)}{(P_R P_C)^2} \right]}{E[A^{hop}]}, \tag{3.20}$$

respectively. The end-to-end cost coefficients, $\mathcal{C}^T$ and $\mathcal{C}^E$, enables quantitative analysis on the cost-efficiency of an anycasting design. In the following sections, we use $\mathcal{C}^T$ and $\mathcal{C}^E$ as the main metrics to capture the end-to-end latency and energy consumption of anycasting designs using different prioritization and wake-up policies.

3.5 Cost-Efficient Prioritization Policy

In this section, we analyze the impact of prioritization policies on the end-to-end cost-efficiency based on the end-to-end cost model presented in Section 3.4. By separating the prioritization policy independent and dependent components in the end-to-end cost coefficient expression, two greedy forwarding metrics are proposed for fully distributed forwarding decision targeting transmission latency and energy cost minimization, respectively.

3.5.1 Determining Cost-Efficient Forwarding Metrics

To evaluate how prioritization policy affects the end-to-end cost coefficient, we first deduce the prioritization policy independent components from (3.20) and (3.19). For (3.19), both T^{ctrl} and T^{data} are fixed values for all sensor nodes. Therefore, they are independent of the prioritization policy in each hop. P_R is the probability of the RTS packet being successfully received by at least one potential receiver. According to (3.6) and (3.9), P_R is given as

$$P_R = \sum_{n=1}^{\rho'} \binom{\rho'}{n} \left[1 - (1 - P_{R0})^n\right] P_{wake}^n (1 - P_{wake})^{\rho' - n}, \tag{3.21}$$

where $\rho' = \frac{\rho}{2}$ is the number of neighboring nodes located within the forwarding region. Therefore, P_R is determined by P_{wake} and ρ, and thus is independent of the prioritization policy. We use I_1^T and I_2^T to represent the prioritization policy independent components as

$$\begin{aligned} I_1^T &= T^{ctrl}/P_R, \\ I_2^T &= T^{data}, \end{aligned} \tag{3.22}$$

and thus the prioritization policy dependent components of $\mathcal{C}^T$ can be determined as P_C, P_D, P_A and $E[A^{hop}]$.

$$\mathcal{C}^T = \frac{1}{E[A^{hop}]} \cdot \left(\frac{I_1^T}{P_C} + \frac{I_2^T}{P_D \cdot P_A} \right). \tag{3.23}$$

To minimize $\mathcal{C}^{\mathcal{T}}$, a series of forwarding decisions need to be made end-to-end based on the prioritization policy to achieve global optimization. However, because of the distributed nature of anycasting operation, the forwarding decision can only be made at each hop with local information. Therefore, in order to minimize the end-to-end latency coefficient, $\mathcal{C}^T$, through distributed anycasting operation, we define

$$\mathcal{F}_i^T = \frac{1}{\frac{I_1^T}{P_{C,i} \cdot A_i} + \frac{I_2^T}{P_{D,i} \cdot P_{A,i} \cdot A_i}} \tag{3.24}$$

as the local forwarding metric used to prioritize the potential receivers, where $P_{C,i}$, $P_{D,i}$, $P_{A,i}$ and A_i are the expected packet reception rates and forwarding advancement achieved at hop i. These parameters can be decided locally if a particular sender and potential receiver pair are given [56, 62]. I_1^T and I_2^T are fixed values if the network use uniform duty-cycle and packet length. These value can be calculated according to 3.22 and 3.21, and pre-configured at each sensor node. Upon receiving the RTS packet, each potential receiver evaluates its priority as the winning receiver using (3.24). The node with the largest $\mathcal{F}_i^T$ value gets the highest priority to become the winning receiver. The difference between (3.19) and (3.24) lies in the scope of optimization [62]. Since anycasting makes the forwarding decision in a fully distributed manner, (3.24) can been seen as a greedy forwarding metric for potential node prioritization that targets the end-to-end latency coefficient minimization.

Similarly, for (3.20), if we represent the prioritization policy independent components using

$$\begin{aligned} I_1^E &= \frac{\varepsilon \cdot T^{ctrl}}{P_R}, \\ I_2^E &= 2\varepsilon \cdot T^{data}, \\ I_3^E &= \frac{\varepsilon \cdot [\frac{1}{2}\rho_I T_{P|R} + P_R(\rho_I^R T_{R\_C} + T_C)]}{P_R^2}, \end{aligned} \tag{3.25}$$

we can then determine prioritization policy dependent components of $\mathcal{C}^E$, as P_C, P_D, P_A and $E[A^{hop}]$, where

$$\mathcal{C}^E = \frac{1}{E[A^{hop}]} \cdot \left(\frac{I_1^E}{P_C} + \frac{I_2^E}{P_D P_A} + \frac{I_3^E}{P_C^2} \right). \tag{3.26}$$

Accordingly, a greedy forwarding metric, $\mathcal{F}_i^E$, is defined as

$$\mathcal{F}_i^E = \frac{1}{\frac{I_1^E}{P_{C,i}\cdot A_i} + \frac{I_2^E}{P_{D,i}P_{A,i}\cdot A_i} + \frac{I_3^E}{P_{C,i}^2\cdot A_i}}. \tag{3.27}$$

3.5.2 Impact of varying forwarding metrics

In this subsection, we demonstrate how different forwarding metrics prioritize the potential receivers differently within the disconnection distance of an anycasting sender based on log-normal channel model using the priority level distribution and contour plot within the forwarding region. The network and channel model parameters used in deriving the analytical results are listed in Table 3.1 to characterize the performance of MicaZ nodes [40] in an indoor environment. Unless otherwise noted, the parameters in Table 3.1 are used for deriving the numerical and simulation results in the rest of the chapter.

Consider a single-hop anycasting operation initiated at sender m, as shown in Fig. 3.2. Upon receiving the preamble sent by m, located at (0,0), any potential receiver within the disconnection distance evaluates its priority of becoming the winning receiver, using $\mathcal{F}$. If a potential receiver i, located at (x,y), is selected as the next-hop (WR), the forwarding advancement A_i achieved by this single-hop forwarding is given by $A_i(x, y) = x$. Therefore, based on the channel model in (3.3), the priority level of potential receiver i can be derived using (3.24) and (3.27) as

$$\begin{aligned}
\mathcal{F}^T(x, y) &= \frac{x}{\frac{I_1^T}{\overline{\Psi}(D,l_C)} + \frac{I_2^T}{\overline{\Psi}(D,l_D)\cdot\overline{\Psi}(D,l_A)}}, \\
\mathcal{F}^E(x, y) &= \frac{x}{\frac{I_1^E}{\overline{\Psi}(D,l_C)} + \frac{I_2^E}{\overline{\Psi}(D,l_D)\cdot\overline{\Psi}(D,l_A)} + \frac{I_3^E}{\overline{\Psi}(D,l_C)^2}},
\end{aligned} \tag{3.28}$$

where $D = \sqrt{x^2 + y^2}$; l_C, l_D, l_A are the expected packet reception rates achieved between the node m and node i.

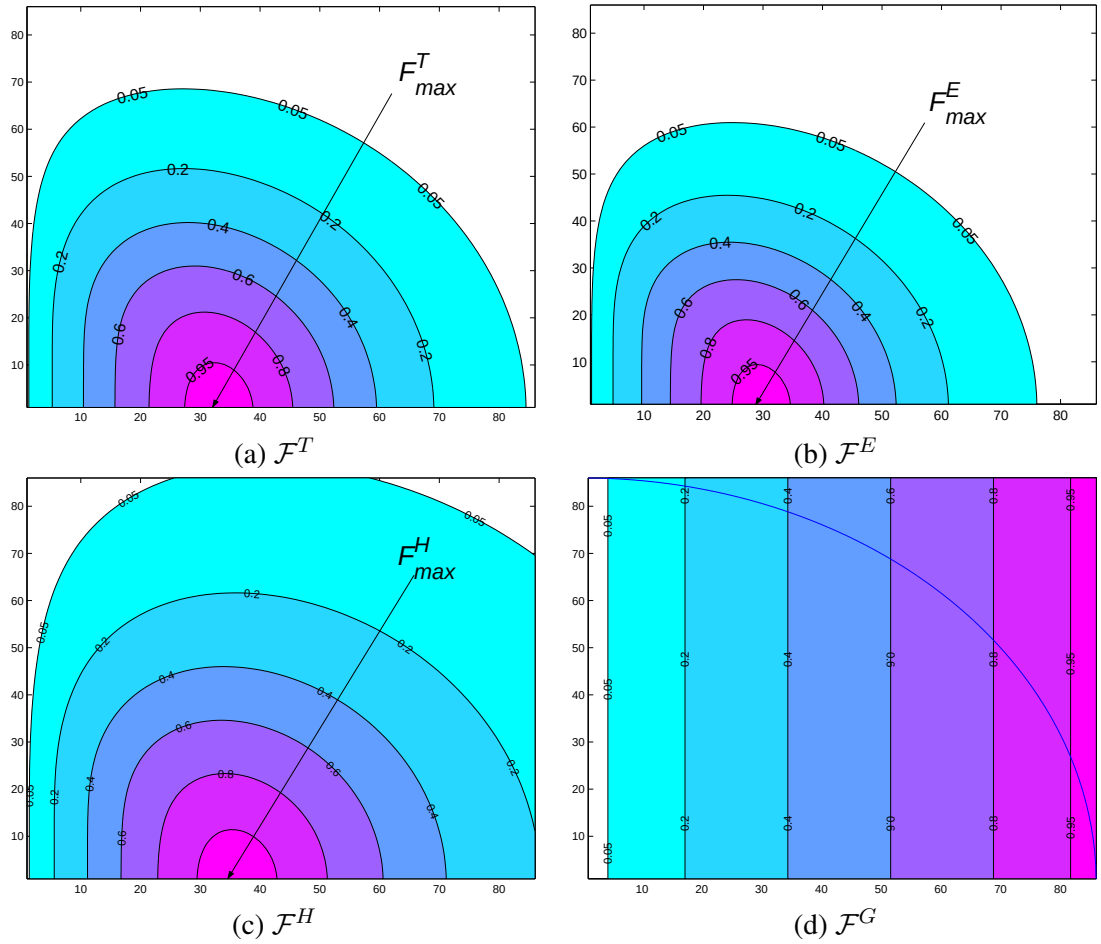
The contour plots of the priority level distribution using the proposed forwarding met-

Table 3.1: Analytical and Simulation Evaluation Settings

Channel Model Parameters	
Radio bandwidth	250 kbps
Path loss exponent η	3
Shadow fading variance σ	4.5
Transmission power P_t	0dBm
Noise power P_n	-100dBm
Reference distance D_0	0.3m
Network Parameters	
RTS/CTS/ACK Packet Length	10 byte
DATA Packet Length	50 byte
Node Degree ρ	40
Duty-cycle d_c	0.1

rics, $\mathcal{F}^T$ and $\mathcal{F}^E$, are shown in Fig. 3.4 (a) and (b) separately based on our analytical model. The contour plots of priority level distribution for a geographic forwarding metric proposed in [33] [55], $\mathcal{F}^G(x, y) = x$, and a heuristic forwarding metric proposed in [56], $\mathcal{F}^H(x, y) = \overline{\Psi}(x, l_D) * x$, are also shown in Fig. 3.4 (c) and (d), respectively, for comparison. We only show the situation where $x \geq 0$ and $y \geq 0$, which is symmetric to the situation when $x \geq 0$ and $y \leq 0$.

From Fig. 3.4, we observe that all channel-quality-aware forwarding metrics, $\mathcal{F}^H$, $\mathcal{F}^T$ and $\mathcal{F}^E$, consider the per-hop cost and forwarding advance (cost-advancement) tradeoff while deciding the winning receiver from the forwarding set. Instead of prioritizing the potential receivers with larger forwarding advancement and lossy wireless link, the channel-quality-aware forwarding metrics give higher priority to the the potential receivers located within the transient region of wireless channel [12]. According to the simulation results given in Fig. 3.7, such prioritization mechanisms help decrease about $60\% \sim 90\%$ the overall end-to-end communication cost compared with $\mathcal{F}^G$. From Fig. 3.4, we also observe that the achieved forwarding metric values can be maximized by a potential receiver located at a point $(0, y_o)$, pointed by the arrow in Fig. 3.4, for different channel-quality-



(a) $\mathcal{F}^T$ (b) $\mathcal{F}^E$

(c) $\mathcal{F}^H$ (d) $\mathcal{F}^G$

Figure 3.4: Contour plots of the priority level distribution (normalized) using different forwarding metrics.

aware forwarding metrics. The value of y_o varies with different forwarding metrics and it is in inverse ratio to the gradient of the distribution contour. This implies that different cost minimization objectives, for example, latency or energy consumption, will lead to different forwarding decisions in considering the cost-advancement tradeoff. Generally, if the per-hop communication cost increases more dramatically with the increasing forwarding advancement, the resulting y_o will decrease with a larger gradient of the distribution contour. The forwarding advancement achieved in single hop forwarding needs to be normalized by a particular cost model so that the greedy forwarding metric can be specified

for particular end-to-end cost minimization.

3.6 Cost-Efficient Wake-Up Policy

According to the end-to-end cost analysis, the preamble length t_P determines the wake-up policy in an anycasting design if the duty-cycle operations are fixed for each sensor node in the network. In order to determine whether waiting for the most suitable neighboring node to wake up always results in overall cost-efficiency, the relationship between the preamble length and the achievable end-to-end cost coefficient needs to be understood. In this section, we first derive the cumulative distribution function (CDF) of end-to-end cost coefficient, $\mathcal{C}$, using the cost-efficient forwarding metrics proposed in Section 3.5. Based on that we show how the expected end-to-end cost coefficient under certain confidence-levels can be controlled by preamble length adaptation under different network conditions.

3.6.1 Distribution of End-to-End Energy and Latency Efficiency with Varying Preamble Length

According to the basic anycasting operation specified in Section 3.3.2, if $\mathcal{F}^T$ is the forwarding metric used in the forwarding decision, $E[\mathcal{F}^T]$ is the expected $\mathcal{F}_i^T$ value achieved by the winning receiver in single-hop anycasting. According to (3.17) and (3.18),

$$E[\mathcal{F}_i^T] = \frac{1}{\mathcal{C}^T} = \frac{D^{e2e}}{E[C^{e2e}]}. \tag{3.29}$$

Since the $\mathcal{F}_i^T$ distribution within the disconnection distance of the sender is known according to (3.28), as shown in Fig. 3.4, the achieved $E[\mathcal{F}^T]$ in a single-hop forwarding must be less than a value a, if no potential receiver exists in the region enclosed by the F^T contour of level a. Therefore, the probability of $E[\mathcal{F}^T]$ being less than a value a is given as

$$P(E[\mathcal{F}^T] \leq a) = \begin{cases} (1 - \frac{A_a}{A})^{\rho_I^R} & 0 \leq a \leq E[\mathcal{F}^T]_{max} \\ 0 & a < 0 \\ 1 & a > E[\mathcal{F}^T]_{max} \end{cases} \tag{3.30}$$

where A is the area of region enclosed by the disconnection distance. $E[\mathcal{F}_i^T]_{max}$ is the maximum achievable $E[\mathcal{F}^T]$ value for a potential receiver located within A. ρ_I^R is the expected number of nodes successfully receiving the RTS packet, i.e. the size of forwarding set, which is given in (3.10). $A(a)$ is the area of region enclosed by the $\mathcal{F}_i^T$ contour at level a, which is represented as an implicit function of x and y. The area of the region enclosed by the contour at level a can then be found using numerical analysis on (3.28).

Based on (3.29), the CDF of end-to-end latency coefficient, $\mathcal{C}^T$, achieved by an anycasting design using F^T as the forwarding metric, is found as

$$P(\mathcal{C}^T < c_t) = 1 - P\left(E[\mathcal{F}^T] \leq \frac{1}{c_t}\right). \tag{3.31}$$

Similarly, we derive the CDF of end-to-end energy coefficient, $\mathcal{C}^E$, achieved by an anycasting design under the forwarding metric, F^E, as

$$P(\mathcal{C}^E < c_e) = 1 - P\left(E[\mathcal{F}^E] \leq \frac{1}{c_e}\right). \tag{3.32}$$

Equations *(3.31) and (3.32) enable us to quantitatively analyze the end-to-end cost-efficiency of an anycasting design with specific design parameters in a statistical manner.* According to the relationship between $E[C^{e2e}]$ and $\mathcal{C}$, as shown in (3.29), a point on a CDF curve of $\mathcal{C}^T$ with *confidence-level*, $L_{con} = 0.8$, and $\mathcal{C}^T = c_t$ indicates that 80% percent of the packets can be delivered with end-to-end latency less than $(c_t \cdot D^{e2e})$. Similarly, a point on the CDF curve of $\mathcal{C}^E$ with *confidence-level*, $L_{con} = 0.6$, and $\mathcal{C}^T = c_e$ indicates that

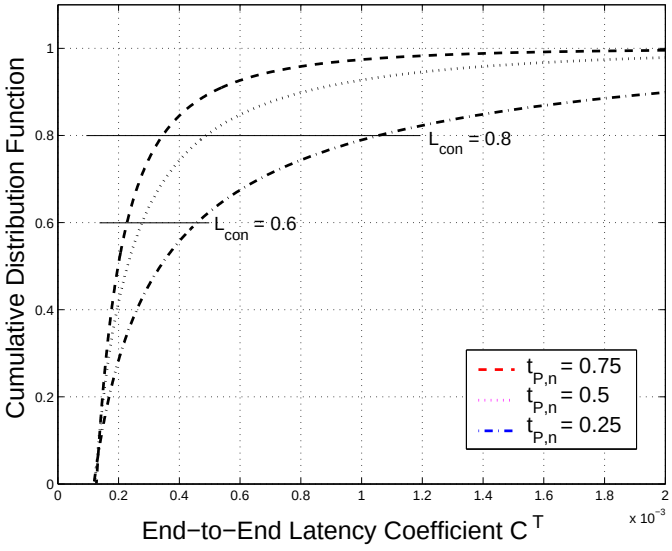
60% percent of the packets can be delivered with end-to-end energy consumption less than $(c_e \cdot D^{e2e})$. If we assume a fixed confidence-level in the end-to-end cost analysis, for a set of CDF curves derived by using different anycasting design parameters, the one resulting in the minimum $\mathcal{C}$ value gives the anycasting design with the most cost-efficient design parameters.

3.6.2 Determining Cost-Efficient Preamble Length

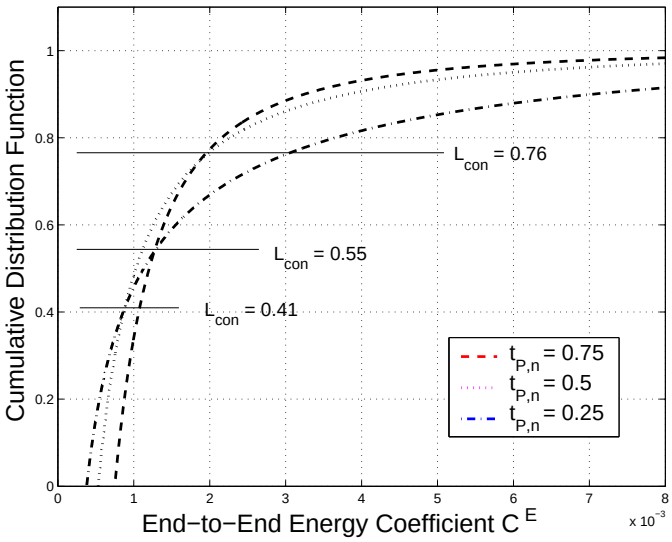
In Fig. 3.5, the sample CDF plots of $\mathcal{C}^T$ and $\mathcal{C}^E$ are shown with varying normalized preamble lengths based on our analytical model. From Fig. 3.5, we find that the end-to-end communication cost can be controlled by varying the preamble length in anycasting design. Based on the analytical results shown in Fig. 3.5, we can make the following observations:

First, a larger normalized preamble length results in better end-to-end latency efficiency. According to Fig. 3.5 (a), by varying the $t_{P,n}$ from 0.75 to 0.25, the achieved $\mathcal{C}^T$ is decreased from 1.4 to 0.35 at $L_{con} = 0.8$, which results in approximately a 4 times increase in end-to-end latency efficiency. The analytical results indicate that although increasing the preamble length introduces extra delay in waking up the neighboring node, the overall transmission delay can be decreased with smaller achievable $\mathcal{C}^T$ value due to a better winning receiver becoming available in a larger forwarding set.

Second, a larger normalized preamble length cannot guarantee better end-to-end energy efficiency. From Fig. 3.5 (b), we observe that, compared with end-to-end latency cost, the achieved end-to-end energy coefficient does not monotonously decrease with increasing preamble length, for example, at $L_{con} = 0.6$. The analytical results indicate that a longer preamble length results in more neighboring nodes being awakened to receive the RTS packet; thus, more energy is consumed at the anycasting receivers. On the other hand, a longer preamble length results in larger ρ_I^R and leads to a higher probability of achieving a specific end-to-end energy coefficient. The tradeoff between waking up more neighboring



(a) CDF v.s. $\mathcal{C}^T$



(b) CDF v.s. $\mathcal{C}^E$

Figure 3.5: Cumulative distribution function of end-to-end latency coefficient $\mathcal{C}^T$ and end-to-end energy coefficient $\mathcal{C}^E$ with normalized preamble length equals to 0.25, 0.5 and 0.75 separately. $d_c = 0.1$.

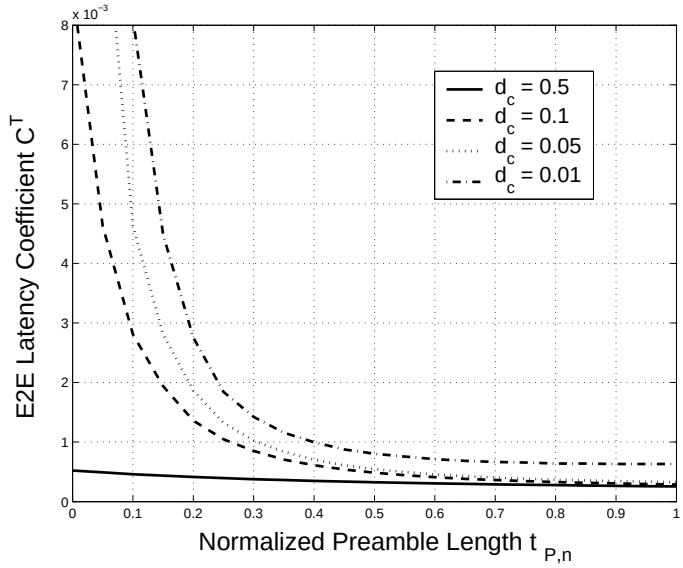
nodes to achieve better $\mathcal{C}^E$ value and consuming more energy in each transmission attempt leads to an optimal $t_{P,n} < 1$ for minimizing the end-to-end energy coefficient.

To fully investigate the relationship between normalized preamble length $t_{P,n}$ and the achieved normalized end-to-end cost coefficient $\mathcal{C}$ in a larger anycasting design parameter space, Fig. 3.6 is plotted based on our cost model to show the situation when an anycasting design operates in high ($d_c = 0.5$), low ($d_c = 0.1$ and $d_c = 0.05$), and extremely low ($d_c = 0.01$) duty-cycled WSNs.

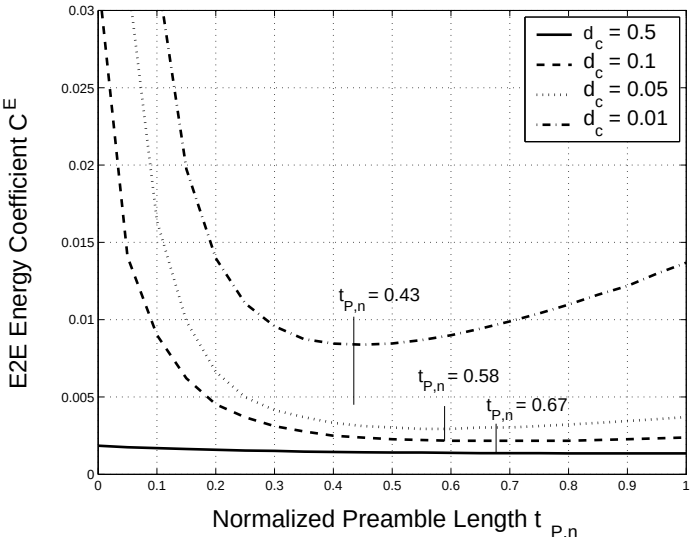
High duty-cycled WSNs ($d_c \simeq 1$): The end-to-end cost-efficiency is not sensitive to the preamble length adaptation. When the normalized preamble length is increased from 0 to 1, both $\mathcal{C}^T$ and $\mathcal{C}^E$ are hardly changed. Therefore, in high duty-cycled MicaZ network, the cost-efficiency of an anycasting design cannot be significantly improved through preamble length adaptation.

Low duty-cycled WSNs ($d_c \simeq 0.1$): The end-to-end cost-efficiency is sensitive to preamble length adaptation, especially when $t_{P,n}$ is relatively small. According to the analytical results in Fig. 3.6, when the normalized preamble length $t_{P,n}$ increases from 0 to 0.4, both end-to-end latency and energy coefficient can be decreased for about 90%. For the scenarios when $d_c = 0.05$ and $d_c = 0.01$, the end-to-end energy coefficient $\mathcal{C}^E$ achieved at a confidence level 0.9 can be minimized with preamble length $t_{P,n} = 0.67$ and $t_{P,n} = 0.58$ respectively. However, both $\mathcal{C}^E$ and $\mathcal{C}^T$ are relatively stable when $t_{P,n}$ changes in $[0.4, 1]$. Accordingly, $t_P \geq 0.4$ is recommended for anycasting design to achieve high end-to-end cost-efficiency in low duty-cycled MicaZ network.

Extremely low duty-cycled WSNs ($d_c \simeq 0.01$): The end-to-end cost-efficiency is extremely sensitive to preamble length adaptation, especially when $t_{P,n}$ is relatively small. According to the analytical results shown in Fig. 3.6, $\mathcal{C}^E$ is minimized with a smaller $t_{P,n}$, before $\mathcal{C}^T$ is minimized while $t_{P,n}$ is increased from 0 to 1. If we denote the $t_{P,n}$ value for $\mathcal{C}^E$ minimization as t^o, both $\mathcal{C}^E$ and $\mathcal{C}^T$ can be decreased dramatically while $t_{P,n}$ increases



(a) $\mathcal{C}^T$ v.s. $t_{P,n}$, ($L_{con} = 0.9$)



(b) $\mathcal{C}^E$ v.s. $t_{P,n}$, ($L_{con} = 0.9$)

Figure 3.6: The relationship between the end-to-end cost coefficient $\mathcal{C}$ achieved at confidence-level $L_{con} = 0.9$ and the corresponding preamble length t_P. The duty-cycle d_c is varied from 0.01 to 0.5.

within $[0, t^o]$. However, for the cases when $t_{P,n}$ increases within $[t^o, 1]$, $\mathcal{C}^E$ increases dramatically with slightly decreased $\mathcal{C}^T$. Such analytical results indicate that the communication cost introduced by waking up neighboring nodes contributes more in the overall energy consumption for end-to-end packet delivery in lower duty-cycled WSNs. Therefore, waiting for the most suitable neighboring node to wake up does not guarantee overall energy-efficiency. Based on the above analysis, $t_P = t^o$ is recommended for anycasting design to achieve high end-to-end cost-efficiency in extremely low duty-cycled network. According to our analytical model, when $d_c = 0.01$, $t^o = 0.43$.

3.7 Simulation Validation

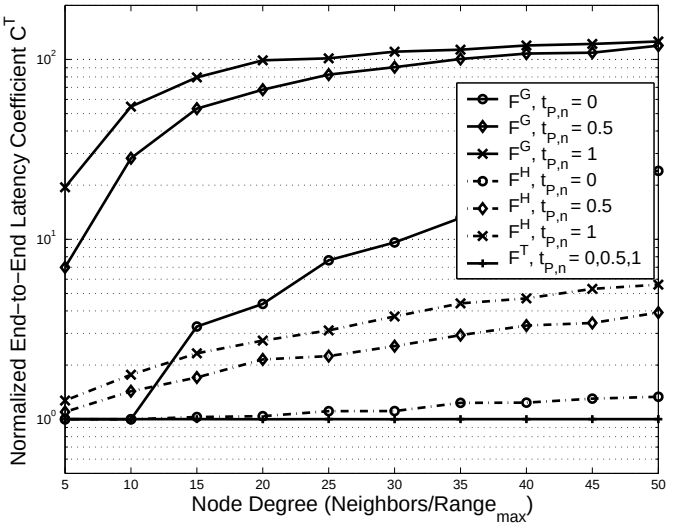
In this section, we present simulation results obtained in wireless network simulator GlomoSim to validate the cost-efficiency of our proposed forwarding metrics and preamble length control guidelines derived based on our cost model. The physical and link layers are designed to follow the log-normal channel model. The anycasting implementation follows the basic anycasting operation described in Section 3.3.2. The simulation parameters remain the same as shown in Table 3.1.

3.7.1 Validation of Cost-Efficient Prioritization Policy

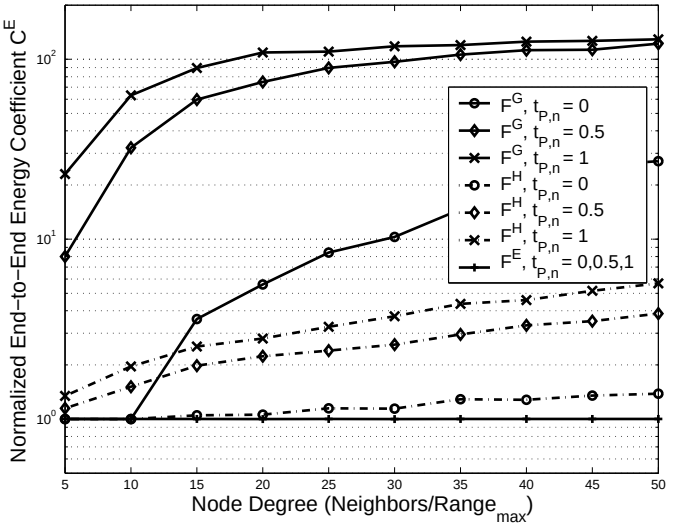
In this simulation senario, the end-to-end communication cost $E[C^{e2e}]$, in terms of transmission latency and network energy consumption, are collected for 5000 packet deliveries from random anycasting senders to the sink using forwarding metric $\mathcal{F}^T$, $\mathcal{F}^E$, $\mathcal{F}^G$ and $\mathcal{F}^H$. The maximum end-to-end transmission distance is $200m$. According to (3.17), the average $(E[C^{e2e}]/D^{e2e})$ values obtained in 5000 packet deliveries are calculated to show the end-to-end cost coefficient achieved by using different forwarding metrics in anycasting operation. Since we focus only on the cost-efficiency variation due to changing forwarding metrics in

an anycasting design, we want to keep the prioritization policy independent parameters of anycasting design to be constant. According to (3.23) and (3.26), P_R, ρ_I and ρ_I^R are the prioritization policy independent components affecting $\mathcal{C}$, and the preamble length t_P is the only anycasting design parameter that affects these values. Therefore, a constant preamble length is used to validate the cost-efficiency of different forwarding metrics. Three simulation scenarios with $t_{P,n} = 0$, $t_{P,n} = 0.5$ and $t_{P,n} = 1$ are constructed respectively to span the entire design space of preamble length.

Fig. 3.7 (a) shows the relative end-to-end latency-efficiency of $\mathcal{F}^G$ and $\mathcal{F}^H$ with respect to $\mathcal{F}^T$ metric, i.e., the $\mathcal{C}^T$ values achieved by metrics $\mathcal{F}^G$ and $\mathcal{F}^H$ are normalized to the latency cost coefficient achieved by $\mathcal{F}^T$. Fig. 3.7 (b) shows the relative end-to-end energy-efficiency of $\mathcal{F}^G$ and $\mathcal{F}^H$ with respect to $\mathcal{F}^E$ metric. The simulation results validate that, compared with geographic forwarding metric $\mathcal{F}^G$, the proposed forwarding metrics, $\mathcal{F}^T$ and $\mathcal{F}^E$, can reduce more than 90% the end-to-end latency and energy consumption, respectively. Compared with heuristic forwarding metric, $\mathcal{F}^H$, $\mathcal{F}^T$ and $\mathcal{F}^E$ can reduce the end-to-end latency and energy consumption for anycasting with long preamble length by about 25%, with moderate preamble length by about 55%, and with short preamble length by about 65%, in the network with moderate node degree. From the simulation results, we can also observe that the cost-coefficient reduction achieved by using the proposed forwarding metrics increase with larger node density or preamble length, i.e., the cost-efficiency is more sensitive to the forwarding metric when the node density is high or the preamble length is large. The reason behind this behavior is that the expected number of potential receivers decreases with smaller node density and preamble length. Accordingly, different forwarding metrics tend to result in the same winning receiver even when the prioritization policy is different. When the expected number of potential receivers approaches 1, the end-to-end cost-efficiency will not be affected by the forwarding metric adopted by an anycasting design.



(a) Normalized Latency Coefficient v.s. ρ



(b) Normalized Energy Coefficient v.s. ρ

Figure 3.7: Comparing the normalized end-to-end latency coefficient $\mathcal{C}^T$ and the energy coefficient $\mathcal{C}^E$ achieved by different forwarding metrics $\mathcal{F}^T$, $\mathcal{F}^E$, $\mathcal{F}^H$ and $\mathcal{F}^G$ with varying network node density.

Table 3.2: Average End-to-End Latency Coefficient (ms/M)

	$d_c = 0.01$	$d_c = 0.1$	$d_c = 0.5$
$t_{P,n} = 0.8$	0.36	0.16	0.17
$t_{P,n} = 0.4$	0.41	0.24	0.17
$t_{P,n} = 0.1$	1.82	0.8	0.19

Table 3.3: Average End-to-End Energy Coefficient (mJ/M)

	$d_c = 0.01$	$d_c = 0.1$	$d_c = 0.5$
$t_{P,n} = 0.8$	56	20.4	9.1
$t_{P,n} = 0.4$	42.5	16.1	8.5
$t_{P,n} = 0.1$	87.7	37.7	10

3.7.2 Validation of Cost-Efficient Wake-Up Policy

The cost-efficient preamble length derived based on our analytical work is validated through simulations by comparing the end-to-end performance of anycasting designs with preamble length $t_{P,n} = 0.1$, $t_{P,n} = 0.4$ and $t_{P,n} = 0.8$. Three simulation scenarios are constructed with $d_c = 0.01$, $d_c = 0.1$ and $d_c = 0.5$ respectively, to capture the case of high, low and extremely low duty-cycled WSN environment. The end-to-end communication cost $E[C^{e2e}]$, in terms of transmission latency and network energy consumption, are collected for 5000 packet deliveries from random anycasting senders to the sink with $D^{e2e} \leq 200m$. We used the average ($E[C^{e2e}]/D^{e2e}$) values obtained in 5000 packet deliveries to show the end-to-end cost-efficiency achieved by anycasting designs with different normalized preamble lengths under different duty-cycle operations.

The simulation results shown in Table 3.2 and 3.3 confirm that,

- For high-duty cycled WSNs, varying the preamble length does not help improve the end-to-end performance of anycasting design significantly.

- For low-duty cycled WSNs, compared with the case when only short RTS packets in sequence are used to wake up the neighboring node, using a normalized preamble

length of $t_{P,n} \geq 0.4$ can improve the performance of anycasting design by reducing about 70% the end-to-end latency and 57% the energy costs.

- For extremely low-duty cycled WSNs, compared with the case when a long preamble or only short RTS packets in sequence are used to wake up the neighboring node, using a normalized preamble at length of around $t_{P,n} = 0.4$ can significantly improve the end-to-end performance of anycasting design by reducing about 51% of the end-to-end energy and 76% the latency costs.

3.8 Summary

In this chapter, we present an end-to-end analysis framework to quantitatively assess the latency and energy efficiency of anycasting operation under log-normal channel model. Based on the proposed analytical model, two greedy forwarding metrics, $\mathcal{F}^T$ and $\mathcal{F}^E$, are proposed to target end-to-end latency and energy consumption efficiency in fully distributed receiver contention process. According to our performance analysis, the proposed forwarding metrics help reduce the end-to-end latency and energy consumption by about 55% for anycasting with moderate preamble length, compared with the existing heuristic forwarding metrics in [56]. Using the proposed forwarding metrics, we further investigate the anycasting wake-up policy design problem. A series of preamble length control guidelines are proposed based on our analytical model to reduce, by more than half, the end-to-end energy and latency cost through tuning the anycasting preamble length in low and extremely-low duty-cycled WSNs.

Chapter 4

LTRES: A Loss-Tolerant Reliable Transport Protocol for Event-Based WSNs

4.1 Introduction

Wireless Sensor Networks (WSNs) are important emerging technologies for providing observations on the physical world with low cost and high accuracy. Accurate and timely observation relies on the collective effort of a large number of sensor nodes. Reliably collecting the data from the sensor nodes to convey the features of a surveillance area, especially the events of interest, to the sink is one of the most critical parts of WSN design. Recently, intensive studies have been carried out at the MAC layer and the network layer to improve the packet transmission performance in terms of throughput and latency against the error-prone and resource-constrained natures of WSNs. However, in order to provide the reliable event sensing guarantee based on application requirements, a transport layer mechanism is highly required for end-to-end performance enhancement on top of the

MAC layer and the network layer operations.

Typically, two kinds of end-to-end reliable data transport requirements can be found in WSN event sensing applications - Loss Sensitive Reliable (LSR) data transport and Loss Tolerant Reliable (LTR) data transport. For LSR, each data packet is required to be successfully transmitted from the source to the destination. Every single packet loss results in a packet retransmission. LSR is commonly required for critical packet delivery, such as query message, control message or event alarm. For LTR, the receiver defines an application-specific reliable data transport requirement for the senders in terms of throughput, loss rate or end-to-end delay. Retransmission is not required for packet loss as long as the application-specific reliable data transport requirements are achieved at the receiver. Most event monitoring applications in WSN require LTR data transport services because collecting sufficient data from the sensor nodes in a timely and energy efficient manner is much more important than guaranteeing the successful reception of each data packet [63].

In this chapter, we propose a distributed data transport protocol to provide Loss-Tolerant Reliable data transport services for dynamic Event Sensing (LTRES) in WSNs. This protocol can be applied to a continuous surveillance WSN with heterogeneous sensing fidelity requirements over different event areas. In LTRES, the sink defines the LTR data transport requirements in terms of required sensing fidelity over an event area. The sensor nodes accordingly adapt their source rates in a distributed manner to meet the LTR requirement based on dynamic network conditions. A loss rate based lightweight congestion control mechanism is used to maintain a low packet loss rate and help the sink determine the satisfiability of an LTR requirement. If an LTR requirement cannot be satisfied by the current network conditions, the sensor nodes can detect the available bandwidth to provide best-effort services using an equation based fair rate control algorithm.

Compared with the existing WSN transport protocols, the main contributions of our work are as follows.

- **LTR transport services with network capacity awareness**: The event-specific LTR transport requirements are defined based on the event goodput observed at the sink and achieved by distributed source rate adaption at the sensor nodes in bounded time, as long as the LTR level is supported by the network capacity. If the event-specific LTR cannot be achieved with current network capacity, the sensor nodes can probe the available network capacity through distributed operations and provide best-effort service. Compared with existing LSR transport protocols, LTRES can provide adaptive end-to-end reliable data transport control based on the dynamic event sensing requirements, which results in minimized packet transmission with less energy consumption. Compared with existing LTR transport protocol ESRT [64], LTRES is able to determine the sustainable LTR transport requirement according to the network conditions and converge to the best-effort service with unsustainable event sensing requirements.

- **Loss rate based congestion detection and avoidance**: Many WSN transport protocols use a buffer occupancy monitoring technique with back-pressure or close loop control to accomplish congestion detection and avoidance. Buffer occupancy monitoring with close loop control introduces extra delay and traffic by sending congestion notifications. Back-pressure can help relief the control overhead, but the reverse routes to the source should be trackable with special routing layer support. Compared with existing work, in LTRES, a light-weigh loss rate based congestion detection and avoidance mechanism is used to alleviate the control overhead. The sink, instead of each sensor node, takes charge of early congestion detection and suppresses the source rates at the aggressive sensor nodes.

- **Event-based fare rate control**: LTRES addresses event-to-sink reliability, where the end-to-end transport reliability requirement is guaranteed for an event area instead

of from each source node. Event-to-sink reliability definition considers the spatio-temporal correlation within the event sensing data [63], but may lead to unfair source rate within the same event area. This could furtherly result in unbalanced energy consumption or biased event data transport. Most existing event-based reliable transport mechanism either uses centralized rate control or AIMD (Additive Increase Multiplicative Decrease) source rate adaptation to address this problem. However, centralized rate control cannot adapt well to the local network dynamics and is up against the scalability problem. AIMD adaptation cannot guarantee a limited convergence time and may cause jittered data rate. In LTRES design, a distributed steady-state throughput estimation approach is used to enforce the fare rate control while providing LTR guarantee with higher bandwidth utilization and fast convergence time.

In Section 4.2, we describe the related work on providing reliable data transport services in WSNs. In Section 4.3, we describe the network model for a common surveillance WSN application and define the reliable data transport requirements for dynamic event sensing. In Section 4.4, we discuss how to achieve the required event sensing fidelity using source rate control and congestion control. Based on that, we introduce the LTRES design, protocol operation and give the protocol correctness and convergence proof. In Section 4.5, we evaluate LTRES in wireless network simulator GloMoSim [50] using different application scenario with different sensing reliability requirements. We also compare the performance of LTRES with other proposed algorithms in the literature in terms of convergence time, achievable reliability level and average packet loss rate. The chapter is concluded in Section 4.6.

4.2 Related Work

Several transport mechanisms have been proposed to provide LSR data transport services over WSNs. RCRT [65] uses traditional TCP-style per-flow rate control with end-to-end packet recovery to ensure flow-based transport reliability. This kind of transport approaches may not be suitable for WSNs due to high retransmission overhead and low convergence speed. Some other approches aim at mitigating the retransmission overhead by using hop-by-hop packet recovery [66] [67] [41]. However, hop-by-hop packet recovery introduces significant control overhead in terms of power and processing. It also requires a large memory space on each sensor node to cache the sent packets for guaranteeing successful retransmission. Moreover, none of the above mechanisms considers network congestion control, which may lead to additional energy consumption on packet loss.

ART [68] improves the traditional LSR design by constructing a coverage set on the sensor network and enforcing end-to-end successful transmission of each event alarm packet from the coverage set to the sink. However, forming the coverage set introduces extra session initialization delay and the alarm-style event detection greatly narrows down its applications.

Since network congestion greatly deteriorates the energy reserves and bandwidth utilization, some transport protocols also focus on exploring new congestion control mechanisms for WSNs. CODA (COngestion Detection and Avoidance) [69] is one of such schemes that provide local congestion detection mechanism by monitoring the local buffer occupancy and the load of each channel. The nodes that detect congestion notify their upstream neighbor nodes to decrease their sending rates through back pressure mechanism. Other similar congestion control mechanisms are proposed in [70][71][72]. Since these mechanisms only focus on congestion control, they do not provide any level of reliable control.

ESRT [64] is the first protocol that provides up-stream LTR transport services along with a congestion control mechanism. The authors introduce event-to-sink transport reliability to address the reliable detection of event features. A centralized closed-loop control mechanism is used to periodically assign each sensor node with a common transmission rate so that a required event sensing fidelity can be achieved at the sink. ESRT also provides a congestion detection mechanism by monitoring the buffer occupancy of the intermediate nodes from an event area to the sink. However, since different sensor nodes may have different local network conditions, the centralized homogeneous rate assignment can deteriorate the overall bandwidth utilization and introduce additional energy consumption due to local congestion. In addition, using the buffer occupancy level of intermediate nodes to determine the congestion level of an entire event area is inaccurate for those sensor nodes not sharing the congestion bottleneck but located within the same event area.

There are some other loss-tolerant data transport protocols proposed recently for WSN applications. The study in [73] focuses on optimizing the sensor node source rate to achieve better network lifetime. The study in [72] tries to guarantee the fair rate control among the sensor nodes based on a pre-constructed tree structure. The study in [58] discusses how to provide end-to-end data transport guarantee in timeliness manner. However, none of them consider the reliability criterion at the transport layer.

4.3 Definitions

4.3.1 Network Model

We consider a homogeneous wireless sensor network with a sensor set $\{S = s_i | i = 1, 2, ..., N\}$ and a sink, where i is the globally unique ID of a sensor node. The sink and the sensor nodes communicate through multi-hop wireless links. Each sensor node transmits source packets at a source rate r_i and forwards any bypass traffic. The sink receives the

source packets from s_i at rate t_i, which is defined as the *per-node goodput*. We consider a common environmental surveillance application, where each sensor node is pre-configured with a common default source rate r_d. r_d can be derived based on prior knowledge of the sensing area and network conditions so that the WSN conducts the sensing with low power consumption and no congestion. Based on the sensing data collected by the sensor nodes, the sink can monitor the sensing field and identify one or more areas of interest, where special events are predicted or detected. We call the area of interest as *event area*, and the sensor nodes covering the event area as *Enode*s, forming an Enode set E. We assume that the sink is able to determine a required event sensing fidelity for an event area based on its computational capability and the dynamic event feature. As a result, the Enodes should adapt their source rates so that enough data associated with the event can be delivered to the sink for reliable event sensing. Whenever an event is deemed uninteresting, the sink can notify the Enodes to set their source rates back to r_d.

4.3.2 Transport Layer Reliability Definition for Dynamic Event Sensing

Providing LTR data transport in WSNs couples accurate event sensing with minimized energy consumption. Therefore, we define the LTR data transport requirements following two aspects: event sensing fidelity and network congestion level. First, we define the event sensing fidelity under our network model.

Definition 1 *Observed Event Sensing Fidelity (OEF_E)*: the observed event goodput achieved at the sink originating from the Enode set E, where

$$OEF_E = \sum_{S_i \in E} t_i$$

OEF_E serves as a simple but adequate event sensing reliability measure at the transport

level [64]. Generally, higher event goodput provides less event sampling distortion; therefore, results in better event sensing fidelity.

Definition 2 *Desired Event Sensing Fidelity (DEF_E)*: the desired event goodput achieved at the sink originating from E, according to the sensing fidelity requirement.

$$DEF_E = E_r$$

DEF_E is determined by the sink based on its computational capability and the event sensing accuracy requirement. Such a decision-making process is application-dependent, which is beyond the scope of our work. Interested readers can refer to [63] for an analysis of this topic.

Definition 3 *Event Sensing Fidelity Level (ESF_E)*: the ratio of observed event sensing fidelity at the sink to the desired event sensing fidelity.

$$ESF_E = \frac{OEF_E}{DEF_E} \tag{4.1}$$

ESF_E reflects the quality of reliable data transport services provided for event sensing. If $ESF_E \geq 1$, the reliable event sensing can be guaranteed by the LTR transport service. If more than one event is identified by the sink, $ESF_E \geq 1$ should be guaranteed for any event area simultaneously to provide LTR services for the WSN under the available network capacity.

From the ESF definition, higher event sensing fidelity means higher event goodput and higher bandwidth requirement. However, a desired event sensing fidelity DEF may not be achievable under the limited wireless channel capacity. Trying to guarantee $ESF_E \geq 1$ without considering the network capacity may lead to network congestion, which not only results in a lower successful packet delivery rate at the sink but more important in energy wasted by the sensing application [69]. Therefore, congestion control is an important aspect

of providing LTR data transport services in WSNs with minimized energy consumption. A congestion control mechanism should be able to dynamically detect the sustainable ESF based on instantaneous network conditions. If such an ESF cannot be supported, the event nodes should explore the upper bound of the network capacity to provide best-effort data transport service.

4.4 LTRES Design

4.4.1 Case Study

In a wireless sensor network, the source rate r_i determines not only the sensing fidelity achieved at the sink, but also the amount of traffic injected into the sensor network [18]. In order to achieve $ESF_E \geq 1$ at the sink, the Enodes have to adapt their source rates properly so that OEF_E can be regulated to approach DEF_E. On the other hand, congestion can be caused or alleviated by increasing or suppressing the source rates of sensor nodes. Therefore finding out the relationship among the source rates, the OEF_E and the network congestion level is critical to our design. A simple simulation scenario is constructed for this purpose using the wireless network simulator GloMoSim [50] with the simulation parameter shown in Table 4.1. The simulation parameters are carefully chosen to reflect typical wireless sensor node capability [40].

The sensing field is uniformly divided into 50 grids. Each sensor node is randomly positioned in a grid. All sensor nodes are pre-configured with $r_d = 1$ pkt/sec. Since sensor nodes are usually static in a surveillance WSN, a proactive routing protocol is selected at the network layer [73]. Two event areas covered by Enode set E_1 and E_2 are separately identified at different locations, where $E_1 = \{s_{36}, s_{37}, s_{46}\}$, $E_2 = \{s_{13}, s_{14}, s_{23}, s_{24}, s_{33}\}$. In order to observe how sensor node source rates affect OEF, which in turn determines the ESF achieved at the sink, all the Enodes uniformly increase their source rates, with event

Table 4.1: Simulation Parameters

Sensing field dimensions	$(100 \times 200)\ m^2$
Sink Location	$(0, 0)$
Number of sensor nodes	50
Sensor node radio range	40m
Packet length	32 bytes
IFQ length	5 packets
Radio Bandwidth	250 kbps
MAC layer	IEEE 802.11

source rate defined as

$$ESR_E = \sum_{s_i \in E} r_i.$$

From Fig. 4.1 (a), we find out that, for E_1, OEF linearly increases with ESR up to a threshold approximately at $ESR = 45$ pkt/sec. After that, with continuously increasing ESR, the linear relationship between OEF and ESR is broken and OEF reaches its upper bound at around 50 pkt/sec. For E_2, the linear relation also holds before ESR reaches a threshold at around $ESR = 65$ pkt/sec; while after that, OEF does not reach its upper bound, but only slow down its increase until ESR reaches the maximum at around 90 pkt/sec.

If we further investigate the loss rates of each Enode, denoted as $l_i = \frac{t_i}{r_i}$, from Fig. 4.1 (b), for E_1, all the Enodes within a event area maintain low packet loss rates before ESR reaches a certain threshold and encounter a dramatically increased packet loss rate after that. However, for E_2, only s_{14}, s_{24} and s_{33} follow the behavior of E_1; while s_{13} and s_{23} remain at a low loss rate until ESR reaches 155 pkt/sec. The different loss rates for different Enodes explain why OEF_{E_2} still increases during $65pkt/sec < ESR < 130pkt/sec$ after the linear relation is broken, compared with E_1. From the above results, we make the following observations:

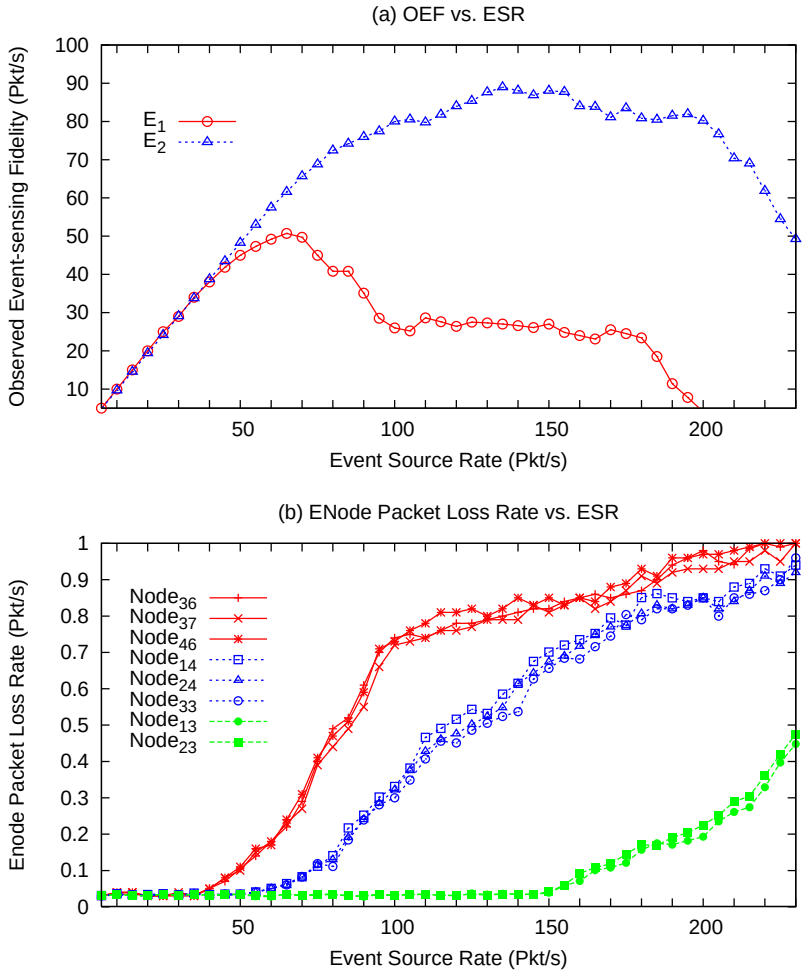


Figure 4.1: Effect of varying the event source rate (ESR) on observed event sensing reliability (OEF) at the sink and the packet loss rates at Enodes. The ID of an Enode is denoted by node number.

Observation 1: *Loss rate can be used as a simple and accurate indication of upstream congestion level of Enodes.*

In WSNs, packet loss is mainly due to two reasons: wireless link error and congestion [69]. When the source rate is low, the traffic load in the network is also low. Only the wireless link error affects the packet transmission; thus a steady low loss rate can be observed at each Enode. When the sensor node source rate is increased beyond a certain threshold value, the traffic load would exceed the network capacity. In this case, both the

wireless link error and the network congestion affect the packet transmission; thus the loss rate dramatically increases at the event nodes that share the congestion bottleneck.

Observation 2: *The network status can be divided into three regions with increasing source rates at Enodes.*

In **Region 1**, OEF increases linearly to ESR with no network congestion. Steady low loss rates can be observed at all Enodes. When network works in this region, with application-specific DEF, the required ESR can be derived through the linear relation. In addition, the maximum energy efficiency can be achieved with lowest packet loss rate. In **Region 2**, higher OEF can be achieved with increasing ESR; however, the linear relation between OEF and ESR is broken due to local network congestion. Dramatically increased loss rates can be observed at certain Enodes sharing a congestion bottleneck. When network works in this region, the required ESR for specific DEF is not linear predictable. In addition, the energy efficiency for packet delivery is deteriorated because of high packet loss rate on congested routes. In **Region 3**, OEF reaches the upper bound or even decreases with increasing ESR. High loss rates are observed at all Enodes because of full network congestion. The network should avoid working in this region because OEF is not controllable through ESR and the energy efficiency for packet delivery is extremely low.

Observation 3: *A centralized source rate assignment mechanism may deteriorate the achievable ESF without congestion.*

From Fig. 4.1, we clearly conclude that different sensor nodes may have different routing paths and different amounts of bypass traffic. Therefore, the data flows originating from different Enodes may face different network conditions. Centrally assigning a uniform source rate for each Enode by the sink without considering the local network conditions, as the mechanism proposed in the ESRT protocol [64], can simplify the protocol operations. However, in order to avoid network congestion, centralized source rate assingment cannot maximize the bandwidth utilization for the event traffic not sharing the congestion

bottleneck. Therefore, the maximum achievable ESF without network congestion will be deteriorated.

4.4.2 Basic LTRES Design

Based on the above observations, a distributed LTR data transport protocol, LTRES, is designed to achieve dynamic event sensing fidelity requirements with congestion control. In LTRES, the sink dynamically detects the event and defines a Enode set, E, that covers the event area. The sink then measures OEF_E and derives ESF_E as the current quality of LTR service level provided for the event sensing and sends it to Enode set, E, covering the event area. Based on this ESF_E notification for the entire event area and the local network congestion level, each Enode adapts its source rate in a distributed manner so that enough event goodput can be delivered to the sink with $ESF_E \geq 1$. Two important design considerations will be discussed in detail as the follows:

- How to conduct the congestion detection and avoidance so that the network stays in **Region 1** with distributively adapting source rate. This is vital to the energy efficiency of protocol operations and to keep the source rate adaptation follow the simple linear relation with OEF_E.

- How to conduct the distributively Enode source rate adaptation so that the $ESF_E \geq 1$ can be achieved at the sink within fast convergence time. Tow kind of source rate adaptation strategy will be discussed under the scenarios where the network is congestion free or network congestion is detected.

Sink-end Congestion Control

In this subsection, we discuss how LTRES conducts the congestion control. Many transport layer protocols designed for WSN applications use a buffer occupancy monitoring tech-

Table 4.2: Mathematical Notation

Network Model Parameters	
source rate of node i	r_i
per-node good-put achieved by node i	t_i
default source rate	r_d
event source rate of Enode set E	ESR_E
Packet Loss Rate Parameters	
probability of packet loss during transmission	$P(L)$
probability of packet loss due to wireless link error	$P(W)$
probability of packet loss due to congestion	$P(C)$
Steady-State Throughput Parameters	
Kth congestion-free duration	D_K
number of source rate adaptation period in D_K	N_K
source rate achieved at the end of D_K	R_K
total number of packets transmitted in D_K	X_K
number of packets transmitted before congestion in D_K	n_K

nique to accomplish congestion detection and avoidance. ESRT uses a closed-loop congestion control mechanism by monitoring the buffer occupancy of the intermediate nodes from the event area to the sink. The event area is deemed to be congested, if any intermediate node between the event area and the sink is congested. Obviously, this is unfair to those sensor nodes not sharing the congested bottleneck but are located within the event area. CODA [69] also uses a buffer occupancy monitoring technique with back-pressure to accomplish the congestion detection and avoidance. However, for a reliable data transport protocol such as LTRES, back-pressure strategy will make the the end-to-end goodput and protocol convergence time hard to predict.

Following Observation 1 derived in Section 4.1, LTRES uses a loss rate based lightweight ACK mechanism to provide congestion control. In our network model, proactive routing is supposed to be used at the network layer. Therefore, the data flows originating from E have static routes. If other types of routing techniques are used in the network, as long as a static route is used for a flow, an static end-to-end wireless path model can be applied to derive the probability of packet loss due to wireless congestion and wireless link error [74],

which is shown as the follows:

$$P(L) = 1 - [1 - P(W)][1 - P(C)] \tag{4.2}$$

where $P(L)$ is the probability of packet loss during transmission; $P(W)$ is the probability of packet loss due to wireless link error; $P(C)$ is the probability of packet loss due to congestion. Since the WSN starts from no network congestion with every sensor node transmitting at r_d, according to (4.2), $P(L) = P(W)$. Therefore, the sink can estimate the path $P(W)$ using a weighted moving average of the instantaneous packet loss rates as

$$Avg[P(W)] = (1 - w_q) * Avg[P(L)] + w_q * Ins[P(L)] \tag{4.3}$$

where w_q reflects the channel diversity. A larger w_q value can be used in a highly dynamic wireless channel and vice versa. The sink periodically observes the loss rate at each Enode using the formula:

$$Ins[P(L)] = \frac{r_i - t_i}{r_i} \tag{4.4}$$

If a steady low loss rate is observed, the upstream routing path for this Enode is deemed to have no congestion or low congestion level; thus $Avg[P(W)]$ is updated according to (4.3). If a dramatically increased loss rate is observed compared with $Avg[P(W)]$, the upstream routing path for this Enode is deemed to be congested. As a result, a congestion notification is sent to the congested event node to trigger the congestion avoidance operation.

Node-end Distributed Source Rate Adaptation

Whenever an event area is identified, the ESF_E is evaluated by the sink and sent to E as an event sensing reliability measure at the transport level. Based on this event sensing

reliability measure, the Enodes periodically conduct the distributed source rate adaptation with network congestion level awareness.

Based on Observation 2, different source rate adaptation strategies are designed for different network conditions. First, if no network congestion is detected, where network stays in Region 1, each Enode performs multiplicative increase (MI) operation on source rate adaptation to approach $ESF_E = 1$ in an aggressive manner. Since ESR is linear to OEF without network congestion, if each Enode satisfies

$$r_i = \frac{r_{i,0}}{ESF_{E,0}},$$

where $r_{i,0}$ is the initial source rate of Enode i and $ESF_{E,0}$ is the initial event sensing fidelity measured at the sink for the entire Enode set E, the distributively source rate adaptation can be stopped with $ESF_E = 1$. The distributively MI source rate adaptation in Region 1 satisfies the LTR requirement with fast convergence time and low control overhead.

If any network congestion is detected by the sink before $ESF_E = 1$ is satisfied, it implies that the MI operation at certain Enodes leads to a local congestion. Under this situation, although the sink may still achieve higher ESF level with increasing source rate, more energy is consumed due to the high packet loss rate. In order to provide energy-efficient source rate control, the LTRES design requires the congested Enodes to start the available bandwidth detection process. A heuristic approach, such as greedy dichotomy with certain dichotomic depth, can be used in distributed source rate adaptation for fast convergent bandwidth detection. After the bandwidth detection operation, the congested Enodes can achieve the maximized bandwidth utilization with upstream congestion avoidance. These nodes then become inactive Enodes, and stop any source rate adaptation operations. The sink derives the new $ESF_{E,0}$ for the rest of the Enodes. These nodes then restart the MI operation in Region 1. If there is no active Enode left in E, all Enodes

stop the source rate adaptation after bandwidth detection operation. In the case, it implies that the application-specific ESF_E requirement cannot be supported under current network condition. Therefore, the LTRES protocol guarantees to provide best-effort service without network congestion.

4.4.3 Improving the Fairness Among LTRES Data Flows

Compared with a centralized rate assignment mechanism, such as the one used in ESRT, a distributed source rate control considers the local network condition at different Enodes so that the overall network bandwidth utility is improved. However, the distributed bandwidth detection algorithm may lead to unfair bandwidth utilization at Enodes sharing the congestion bottleneck. One possible solution for fairness control among LTRES data flows is using AIMD (Additive Increase Multiplicative Decrease) source rate adaptation, which inherently results in a fair bandwidth assignment among the Enodes sharing the congestion bottleneck [75]. However, AIMD source rate adaptation cannot convergent to a steady throughput. As a result, it will cause a jittered event goodput at the sink, which is not suitable for event sensing applications with specific ESF requirement. In addition, since the resulting goodput is jittered for the Enodes sharing the congestion bottleneck, it is impossible to set an updated ESF requirement for the rest active Enodes to target guaranteed the LTR service through Region 1 operations.

In order to achieve a fair rate control with steady event goodput, LTRES design uses a loss-rate based steady-state throughput estimation algorithm at congested Enode for the bandwidth detection. The steady-state throughput is derived through a congestion-free throughput model for wireless channel [74] with the assumption that the AIMD operation is used in source rate adaptation for bandwidth detection.

Assume an LTRES flow originating from an Enode starting at $t = 0$ transmits $X(t)$

packets in t time period, the steady-state throughput T for this flow is defined as

$$T = \lim_{t \to \infty} \frac{\overline{X}(t)}{t},$$

where $\overline{X}(t)$ is the expected number of packets transmitted in each t time period when the number of t period becomes infinite. Assume that each Enode sharing the congestion bottleneck conduct the bandwidth detection by periodically increasing its source rate additively and decreasing its source rate by half if any congestion is detected at the sink. We call the time period between two source rate adaptation operations as *Source-rate-Adaptation Periods*, SAP. Each Enode keeps the same source rate within a source rate adaptation period. After certain rounds of AI operations are conducted at the congested Enode, a network congestion is detected by the sink. We define the time period between any two congestions as *congestion-free duration*, D_K, and the number of source rate adaptation periods within D_K as N_K. If the total number of packets transmitted in D_K is known as X_K, the throughput achieved within a congestion-free duration can be derived as

$$T(t) = \frac{X_K}{D_K}.$$

For each congestion-free period D_K ending with X_K number of packets sent, the evolution of source rate adaptation can be assumed to be Markov regenerative process [74] with rewards X_K. Therefore, from the renewal theorem, the steady-state throughput T achieved by the AIMD operation for bandwidth detection can be found as

$$T = \frac{\overline{X}}{\overline{D}}, \tag{4.5}$$

where $\overline{X}$ and $\overline{D}$ are the expectation of X_K and D_K, while the number of congestion-free

duration is infinity.

$$\overline{D} = SAP \cdot \overline{N}. \tag{4.6}$$

If we denote the source rate at the end of D_K as, R_K, from the definition of AIMD operations, we know that

$$R_{K+1} = \frac{R_K}{2} + N_K - 1$$

Hence, the expectation of i.i.d. random variable R can be expressed as

$$\overline{R} = 2\big(\overline{N} - 1\big) \tag{4.7}$$

Since we can get X_K from the sum of the packets transmitted in a congestion-free duration D_K,

$$X_K = \frac{1}{2}(R_K + \frac{R_{K-1}}{2}) \cdot N_K \cdot SAP \tag{4.8}$$

Based on (4.7) and (4.8), the expectation of X_i, $\overline{X}$, can be expressed through the substitution of mutually independent random variables, N_K and R_K, where

$$\overline{X} = \frac{3\overline{N}(\overline{N} - 1) \cdot SAP}{2} \tag{4.9}$$

On the other hand, in a congestion-free duration, we assume n_K packets are transmitted before the congestion is detected at the sink. Since the network congestion requires one SAP to be detected and notified to the Enode, $R_K \cdot SAP$ more packets are sent after the

packet loss due to congestion. Hence, $X_K = n_K + R_K \cdot SAP$. Accordingly,

$$\overline{X} = \overline{n} + \overline{R} \cdot SAP \tag{4.10}$$

From (4.5), (4.6), (4.7), (4.9), and (4.10), we obtain the steady-state throughput as

$$T = \frac{1}{4}\left(1 + \sqrt{1 + \frac{24\overline{n}}{SAP}}\right) \tag{4.11}$$

Since n_K gives the number of packets transmitted until a congestion occurs, it is geometrically distributed with the unconditional probability of packet loss due to congestion $P(C)$. According to (4.2),

$$\overline{n} = \frac{1 - P(W)}{P(L) - P(W)} \tag{4.12}$$

Whenever a local congestion is detected by the sink, the congested Enodes will use (4.11) as the best-effort source rate to achieve better overall bandwidth utilization without congestion.

4.4.4 LTRES Protocol Operation

In this subsection, we describe the LTRES protocol operation in details. In LTRES protocol, the Enodes operates mainly in two stages. In Stage One, the un-congested Enodes try to provide the guaranteed LTR service by linear source rate adaptation in Region 1. In Stage Two, the congested Enodes try to provide the best effort service by source rate adaptation using steady-state throughput estimation. Fig. 4.2 gives a flow chart of the LTRES protocol operations. The operational stages are shown in blocks. The sink-end congestion control operations and the control packet exchanged between the sink and Enodes are shown between two operational stages. If LTRES protocol operations finish in Stage one,

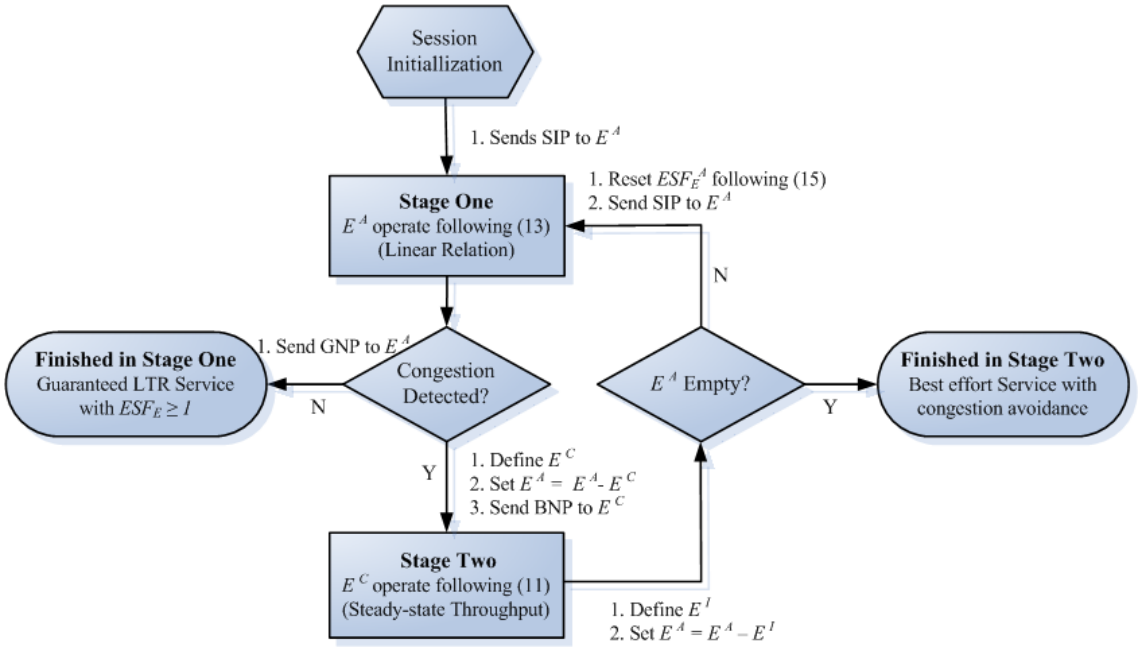


Figure 4.2: Flow chart of the LTRES protocol operations. The operational stages are shown in blocks. The sink-end congestion control operations and the control packet exchanged between the sink and Enodes are shown between two operational stages.

the LTR service is guaranteed with $ESF_E \geq 1$. If LTRES protocol operations finish in Stage Two, the best effort service is provided with congestion avoidance.

Session Initialization Phase

The LTRES operation starts with all sensor nodes operating at default source rate $r_i = r_d$ with no congestion. Whenever an event area is identified by the sink, the sink determines the Enode set E and DEF_E for the event area. It initializes the ESF_{E^A} following Definition 3, sets *Active Enode Set* $E^A = E$, sets *Inactive Enode Set* $E^{IA} = \emptyset$, sets *Congested Enode Set* $E^C = \emptyset$, and sets *Wireless Channel Loss Rate* $P_{i,0}(W) = Avg[Pr_{i,0}(L)]$, where $Avg[P_{i,0}(L)]$ is the average packet loss rate at the beginning of initialization phase. The sink starts the service session by multicasting the Session Initialization Packet (SIP) to E. SIP contains the sequence number, ESF_{E^A} and the E^A ID group.

Stage One (Guaranteed LTR service with congestion control)

Upon receiving the SIP, each active Enode starts the source rate adaptation in $Stage\ One$. In this stage, the active Enodes assume the LTR service, i.e. $ESF_{E^A} = 1$, can be provided through only source rate adaptation in Region 1. Each active Enode $e_i^A \in E^A$ adapts its source rate as follows:

$$r_{i,K+1} = \frac{r_{i,K}}{ESF_E}. \tag{4.13}$$

Meanwhile, it sets the SIP sequence number in the transport header of upstream data packets as an implicit acknowledgement SIP_ACK.

Upon receiving the SIP_ACK from e_i^A, the sink estimates the instantaneous packet loss rate $Ins[P_{i,K}(L)]$ for every W_L using (4.4). W_L is a sliding time window big enough to collect the packet loss rate. In our protocol design, $W_L \geq \frac{100}{ESR}$.

If $Ins[P_{i,K}(L)] - P_{i,K-1}(W) \leq \varepsilon$, the sink updates $Avg[P_{i,K}(W)]$ using (4.3) and multi-casts the Good News Packet (GNP) to e_i^A containing a sequence number. ε is the tolerable variation of loss rate without congestion, which can be derived empirically based on application-specific congestion tolerance level.

Upon receiving the GNP, e_i^A finishes the $Stage\ One$ operation and piggyback the GNP sequence number in the transport header as the implicit GNP_ACK. Upon receiving the GNP_ACK from e_i^A, the sink stops sending GNP to this Enode. If the sink receives GNP_ACK from all Enodes in e_i^A, LTRES operation successfully achieves the LTR event sensing requirements at $Stage\ One$.

Stage Two (Available bandwidth detection with fair rate control)

If $Ins[P_{i,K}(L)] - P_{i,K-1}(W) > \varepsilon$, the upstream path for node i is assumed to be congested. Therefore, the sink identifies node i as Congested Enode e_i^C, and puts it into E^C. The sinks

then multi-casts the Bad News Packet (BNP) to e_i^C containing a unique sequence number, $Ins[P_{i,K}(L)]$, $P_{i,K}(W) = P_{i,K-1}(W)$ and SAP_i, where SAP_i is twice the average end-to-end transmission delay of e_i^C observed in the last W_L.

Upon receiving BNP, e_i^C is assumed not to be able to achieved target source rate in Region 1. In order to avoid the network working in Region 2 and Region 3, the congested Enodes start the $Stage\ Two$ operation. In this stage, e_i^C adapts its source rate following (4.11) and (4.12) using the congestion level information contained in BNP, where $SAP = SAP_i$, $P(L) = Ins[P_{i,K}(L)]$ and $P(W) = P_{i,K}(W)$. It also piggyback the BNP sequence number in its transport header as an implicit acknowledgement BNP_ACK.

Upon receiving the BNP_ACK from e_i^C, the sink updates the $Ins[P_{i,K}(L)]$. Note that, the steady-state throughput model cannot guarantees $Ins[P_{i,K}(L)] - P_i(W) = P_i(C) \leq \varepsilon$. According to [74], the model gives adequate estimation for achieving $P_i(C)$ less than 0.15. As we mentioned in Section 4.3, the steady-state throughput estimation algorithm is used to provide fast convergence time towards the maximum source rate in Region 1 operation and fair rate control among the LTRES flows sharing the congestion bottleneck. Therefore, if $Ins[P_{i,K}(L)] - P_i(W) \leq \varepsilon$ is observed at the sink, the application-specific tolerable variation of loss rate without congestion is satisfied. The sink then identifies node i as the Inactive Enodes and places it into E^I set. All inactive Enodes finish their source rate adaptations to maximize their throughput without congestion.

If $Ins[P_{i,K}(L)] - P_i(W) > \varepsilon$ is observed at the sink, the source rate adaptation at node i is assumed to converge to the steady-state throughput but cannot satisfied the application-specific tolerable variation of loss rate. In this case, LTRES conducts a 3-step bandwidth detection process with Detection Depth $DD = M$.

- Step 1: The sink sends Bandwidth Detection Packet BDP, containing a unique sequence number and $Ins[P_{i,K}(L)]$, to e_i^C for finer bandwidth detection.

- Step2: Upon receiving BDP, e_i^C conducts source rate adaptation following

$$r_{i,K+1} = r_{i,K} \cdot (1 - Ins[P_{i,K}(L)]), \tag{4.14}$$

  and piggybacks the BDP sequence number in its transport header as an implicit acknowledgement BDP_ACK.

- Step3: Upon receiving the BDP_ACK from e_i^C, the sink sets $DD_i = DDi - 1$. If $DD_i = 0$, the source rate adaptation is finished at e_i^C for finer bandwidth detection. The sink identifies node i as the Inactive Enodes and places it into E^I set. if $DD_i \geq 0$, the sink updates the $Ins[P_{i,K}(L)]$, and goes to Step 1.

(4.14) enables e_i^C to drop the source rate r_i in a small percentage based on the instantly detected network capacity, which results in a lower packet loss-rate. DD determines the bandwidth detection depth. Larger DD can achieve a r_i closer to the maximum source rate without congestion; while introduce longer convergence time. According to our simulation results, which are given in Section 5, $M = 2$ can satisfy $\varepsilon \leq 0.05$ for all simulation scenarios.

When all congested Enodes finish the bandwidth detection process, $E^C = \emptyset$. The sink then sets $E^A = E^A - E^I$. If $E^A \neq \emptyset$, the sink updates ESF_E as follows:

$$ESF_{E^A} = \frac{\sum_{s_i \in E^A} t_{i,0}}{DEF_E - \sum_{s_i \in E^I} t_i} \tag{4.15}$$

The sink generates and sends the new SIP with new ESF_{E^A} to E^A. Upon receiving the new SIP, an active Enode e_i^A starts the source rate adaptation from $Stage\ One$.

If $E^A = \emptyset$, all Enodes have gone through the Stage Two bandwidth detection and the steady-state throughput is achieved without congestion. The best-effort service is provided through LTRES protocol operations.

Session Finalization Phase

Whenever the event area is deemed uninteresting by the sink, the sink sends the Session Close Packet (SCP) to E. All Enodes set $r_j = r_d$. The LTRES operation finishes.

4.4.5 Protocol Operation Correctness and Convergence Proofs

In this subsection, two proofs are given to show how the two-stage protocol operation guarantees the sustainable reliable event transport requirements with limited convergence time.

Theorem 1: If LTRES finishes at $Stage\ One$, the LTR service is guaranteed with $ESF_E = 1$.

Proof:

If LTRES finishes at $Stage\ One$, any active Enode e_i^A stops its source rate adaptation before any congestion is detected. Therefore r_i and t_i maintain a linear relationship, i.e., $t_i = k_i \times r_i$. Since e_i^A stops source rate adaptation with $r_i = \frac{r_{j,0}}{ESF_E}$ (4.13), we have

$OEF_{E,STOP} = \sum_{e_i \in E^A} t_{i,STOP} + \sum_{e_i \in E^I} t_{i,STOP}$

$= \frac{\sum_{e_i \in E^A} r_{i,0} \times k_i}{ESF_{E^A}} + \sum_{e_i \in E^I} t_{i,STOP}$

(linear property)

$= \frac{(DEF_E - \sum_{e_i \in E^I} t_i) \sum_{e_i \in E^A} r_{i,0} \times k_i}{\sum_{e_i \in E^A} t_{i,0}} + \sum_{e_i \in E^I} t_{i,STOP}$

(according to (4.15))

$= DEF_E$

Therefore, $ESF_{E,STOP} = \frac{OEF_{E,STOP}}{DEF_E} = 1$.

Theorem 2: LTRES operation converges within $N * (2 + DD)$ source rate adaptation periods, where N is the size of Enode set.

Proof:

(i) Any active Enode e_i^A adopts MI source rate adaptation in $Stage\ One$ operations following (4.13). Therefore LTRES finishes each $Stage\ One$ operation within one source rate adaptation period.

(ii) Any congested Enode e_i^C finishes $Stage\ Two$ operation within at most $DD + 1$ source rate adaptation periods for steady-state throughput estimation and possible 3-step bandwidth detection process. All congested Enodes that have gone through the $Stage\ Two$ operations become Inactive Enode. The Inactive Enodes finish the distributively source rate adaptation for one LTRES session.

(iii) Any e_i^A can become e_i^C and enter $Stage\ Two$ for at most one time throughout a LTRES session.

Based on (i) - (iii), any Enodes must complete the LTRES operation, in terms of distributively source rate adaptation, within $N * (2 + DD)$ source rate adaptation periods.

4.5 Performance Evaluation

In order to study the performance of the LTRES protocol, we once again construct a simulation environment. The sensing field dimensions is $200 * 200m^2$ with 100 sensor nodes. The sink is located at (100,100). Other simulation parameters keep the same as in the case study, which are shown in Table 4.1. Two kinds of event sensing application scenarios, single event with varying DEF requirements and multiple event occurrences, are provided in the the simulation. We also compare LTRES with existing lTR transport protocol ESRT in terms of convergence time, average packet loss rate, and bandwidth utilization. Based on the simulation results, LTRES is shown to converge faster in all simulation scenarios. In addition, LTRES also provide lower packet loss rate and better bandwidth utilization, especially for a high DEF requirements and multiple event occurrences.

4.5.1 Single Event with Varying DEF Requirements

In this subsection, we conduct a simulation with single event and varying DEF requirements to compare the performance of LTRES and ESRT in operation convergence time, overall bandwidth utilization and packet loss rate. We assume the sink identifies an event centered at location (60,40), and covered by $E1 = \{s_{11}, s_{12}, s_{21}, s_{22}, s_{31}, s_{32}, s_{41}, s_{42}\}$. In Scenario I, a low event sensing fidelity requirement is set to $E1$, where $DEF_{E1} = 50$ pkt/s. In Scenario II, the desired event sensing fidelity requirement is increased to a medium level, where $DEF_{E1} = 90$ pkt/s. In Scenario III, the highest event sensing fidelity requirements is set as $DEF_{E1} = 150$ pkt/s. According to the network conditions, we set $r_d = 1$ pkt/s, $\varepsilon = 0.05$, $w_q = 0.5$ as the default protocol parameters for LTRES and $Decision\_Interval = 5s$ for ESRT [64]. These parameters keep the same in the following simulations.

Fig. 4.3 shows the ESF level and packet loss rate trace for LTRES and ESRT in Scenario I, where $DEF_{E1} = 50$ pkt/s. From the ESF trace in each source rate adaptation period, as shown in Fig. 4.2(a), we can find out that LTRES provides LTR service through only $Stage\ One$ operation. ESRT protocol also converges in two SAPs but with longer convergence time. LTRES converges faster than ESRT in most simulation scenarios because LTRES can adjust the SAP with varying W_L; while ESRT sets the SAP to a fixed value without considering the packet source rate. From the average packet loss rate trace in each source rate adaptation period, as shown in Fig. 4.2(b), we can find out that both LTRES and ESRT are able to maintain a steady low loss rate, which is due to wireless link error. Both protocols can converge fast in Scenario I because the low DEF requirement does not result in high event source rate. Therefore, the network keeps work in Region 1 with low packet loss rate and high energy efficiency.

Fig. 4.4 shows the ESF level and packet loss rate trace for LTRES and ESRT in Scenario II, where $DEF_{E1} = 90$ pkt/s. Compared with Scenario I, this simulation scenario

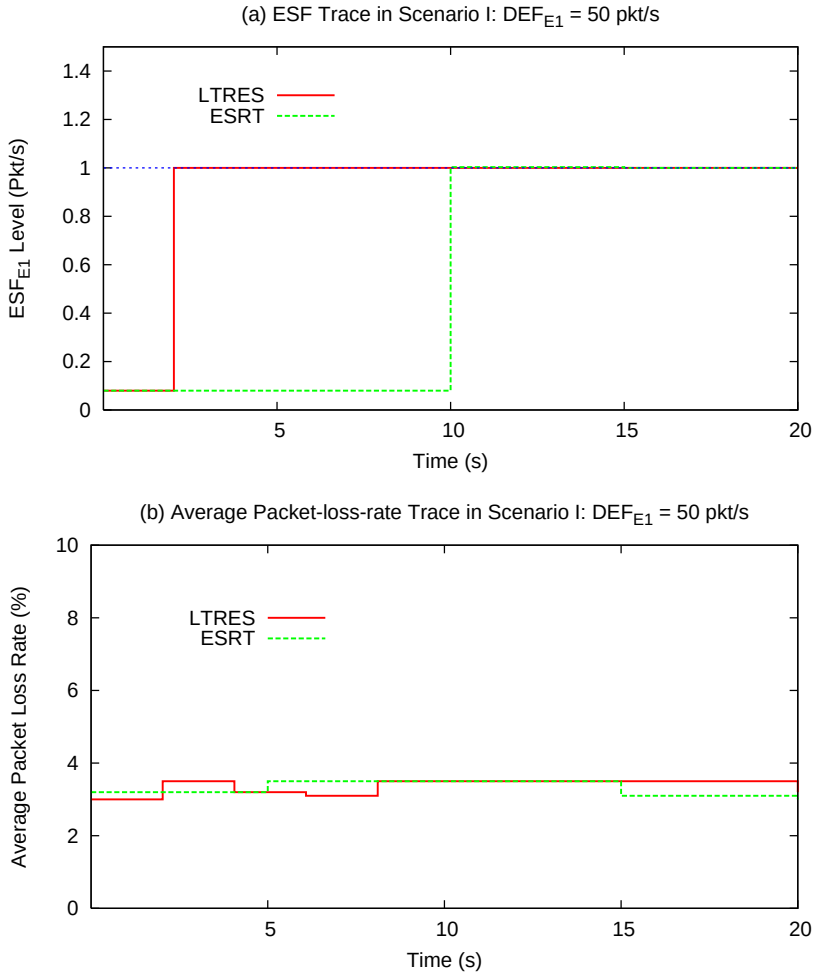


Figure 4.3: ESF level and average packet loss rate trace for LTRES and ESRT protocol with event sensing fidelity requirements in Scenario I $DEF_{E1} = 50$pkt/s. LTRES can achieve required ESF level in Scenario I through only Stage One operations.

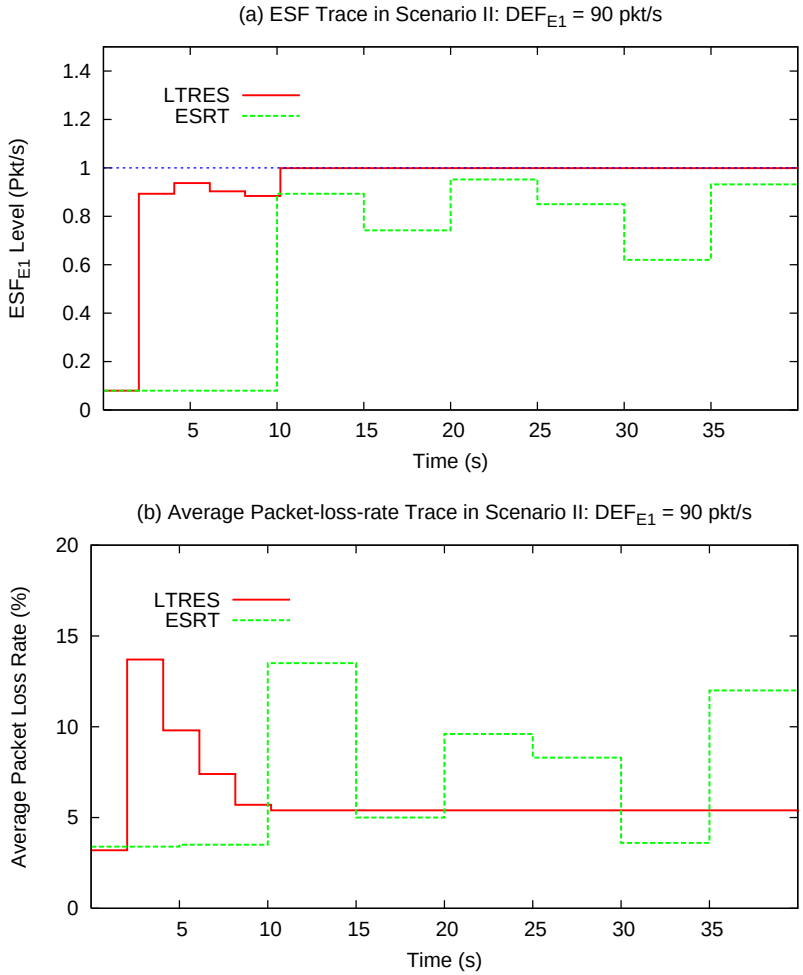


Figure 4.4: ESF level and average packet loss rate trace for LTRES and ESRT protocol with event sensing fidelity requirements in Scenario II $DEF_{E2} = 90$pkt/s. LTRES can achieve required ESF level in Scenario II through both Stage One and Stage Two operations.

requires a higher DEF with higher traffic load and higher overall bandwidth utilization. As shown in Fig. 4.4 (a), LTRES is able to achieve $ESF_E = 1$ through both $Stage\ One$ and $Stage\ Two$ operations. In the first SAP, LTRES protocol tries to meet the ESF requirement through Stage One operations; however, the resulting high ESR leads to a local network congestion, shown as a dramatically increased packet loss rate in Fig. 4.4 (b). In order to provide the LTR service with congestion avoidance, in the next three SAPs, the congested Enodes start the Stage Two operations for steady-state throughput detection with detection depth $DD = 2$. In the last SAP, the sink derives a new ESF requirement, based on which the remaining active Enodes conduct the source rate adaptation through Stage One operations and provide the guaranteed LTR service. However, for ESRT, since it uses a centralized source rate control mechanism, the local congestion detected in itd third SAP triggers the source rate decrease at each Enodes, while the required ESF is not satisfied. Since, only a portion of the Enodes obtaining full bandwidth utilization, ESRT cannot provide the LTR service in Scenario II, but bounds between the states of Congestion/low-ESF and Un-Congestion/Low-ESF, as shown in Fig. 4.4. As a result, ESRT fails to converge in Scenario II with low bandwidth utilization and high average packet loss rate.

For Scenario III, a high DEF is determined by the sink for $E1$. As shown in Fig. 4.5 (a), both LTRES and ESRT cannot provide the guaranteed LTR service in this scenario because the required DEF exceeds the current network capacity. LTRES starts from Stage One operations with all Enode detected to be congested. Therefore, after 3 SAPs, all Enodes finish Stage Two operations and provide the best-effort service for $E1$ with approximately $ESF_E = 0.61$. From Fig. 4.5 (b), the Stage Two operations can provide fast convergence time with congestion avoidance. For ESRT, since it cannot the determine sustainable DEF and provide proper bandwidth detection based on the current network condition, it fails to converge in Scenario III with high packet loss rate and low bandwidth

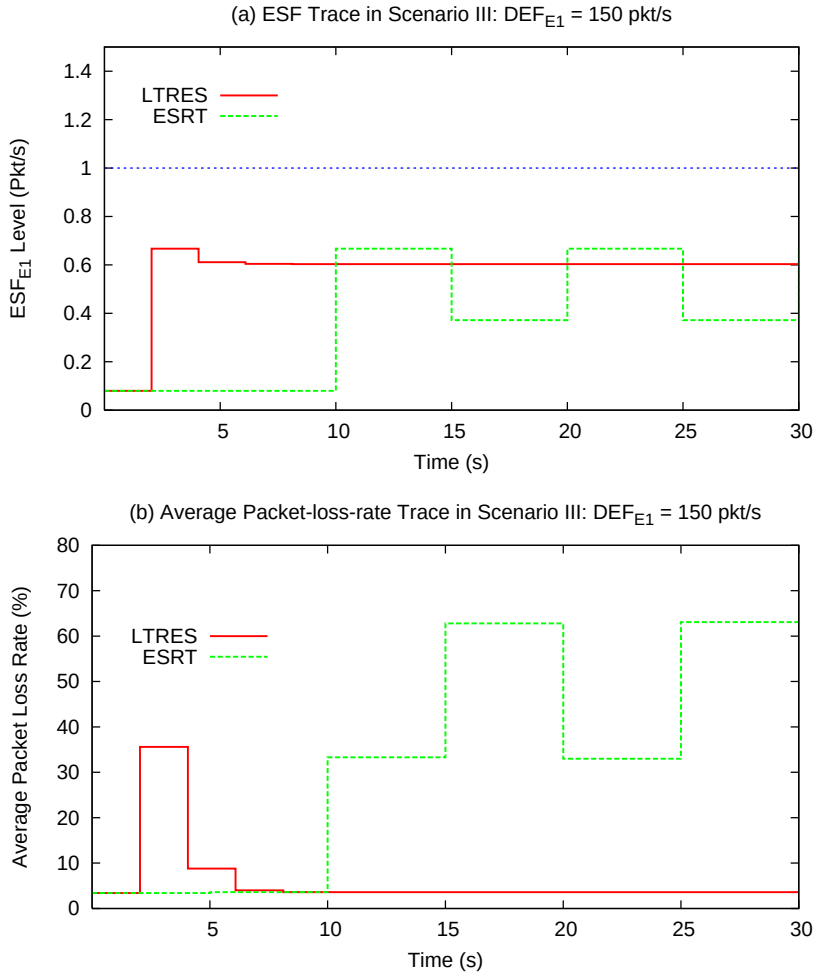


Figure 4.5: ESF level and average packet loss rate trace for LTRES and ESRT protocol with event sensing fidelity requirements in Scenario I $DEF_{E3} = 150$pkt/s. LTRES can provide best-effort service with congestion avoidance in Scenario III ending with Stage Two operations.

utilization.

4.5.2 Providing LTR Service For Multiple-Event Occurrences

In this section, we show how LTRES deal with multiple-event occurrences and compare its performance with ESRT protocol. For demonstration purpose, we show the scenario that three events are detected by the sink simultaneously. We assume that the sink identifies the events centered at location (80,40), (140,20) and (140,160) separately. The first event

is covered by $E_1 = \{s_{13}, s_{14}, s_{23}, s_{24}\}$, with the desired event sensing fidelity requirement as $DEF_{E_1} = 50$ pkt/s. The seconde event is covered by $E_2 = \{s_6, s_7, s_{16}, s_{17}\}$, with the desired event sensing fidelity requirement as $DEF_{E_2} = 40$ pkt/s. The third event is covered by $E_3 = \{s_{76}, s_{77}, s_{78}, s_{86}, s_{87}\}$, with the desired event sensing fidelity requirement as $DEF_{E_2} = 50$ pkt/s. E_1 and E_2 are located close to each other and compete for the channel in the end-to-end routing path. E_3 is located at the opposite corner in the sensing field to E_1 and E_2, thus has isolated routing path.

Fig. 4.6 shows the ESF and average event packet loss rate trace for LTRES and ESRT protocol in each source rate adaptation period. Based on the simulation results, we show that when multiple events exist in the network simultaneously, LTRES perform the same operations as for the single event scenarios. In the first SAP, LTRES attempts to provide the guarantee the LTR service for all events. For E_3, since its route is isolated from both E_1 and E_2, it performs the same as in the single event scenario, where all the Enodes in E_3 finish the protocol operations in Stage One and provide the guaranteed LTR service, as shown in Fig. 4.6 (c) and (f). Since E_1 and E_2 are closed to each other, they compete for the channel in their end-to-end routes. Compared with the single event scenario, the sustainable DEF requirements for both events are decreased. As a result, after Stage One operations, a dramatically increase packet loss rate is observed at the sink and trigger the Stage Two operation at the congested Enodes in E_1 and E_2, as shown in Fig. 4.6 (a) and (d). Since all Enodes in E_1 are detected to be congested after the first SAP, E_1 ends the Stage Two operations in $DD + 1$ SAPs and provides the best-effort service. For E_2, only part of the Enode sharing the congestion bottleneck with E_1. Therefore, after the congested Enodes finish the Stage Two operations in $DD + 1$ SAPs, the active Enodes in E_2 achieve the new ESF requirement set by the sink in fifth SAP and provide the guaranteed the LTR service through Stage One operations, as shown in Fig. 4.6 (b) and (e).

While for ESRT protocol, because of its centralized ESR adaptation policy, it has to

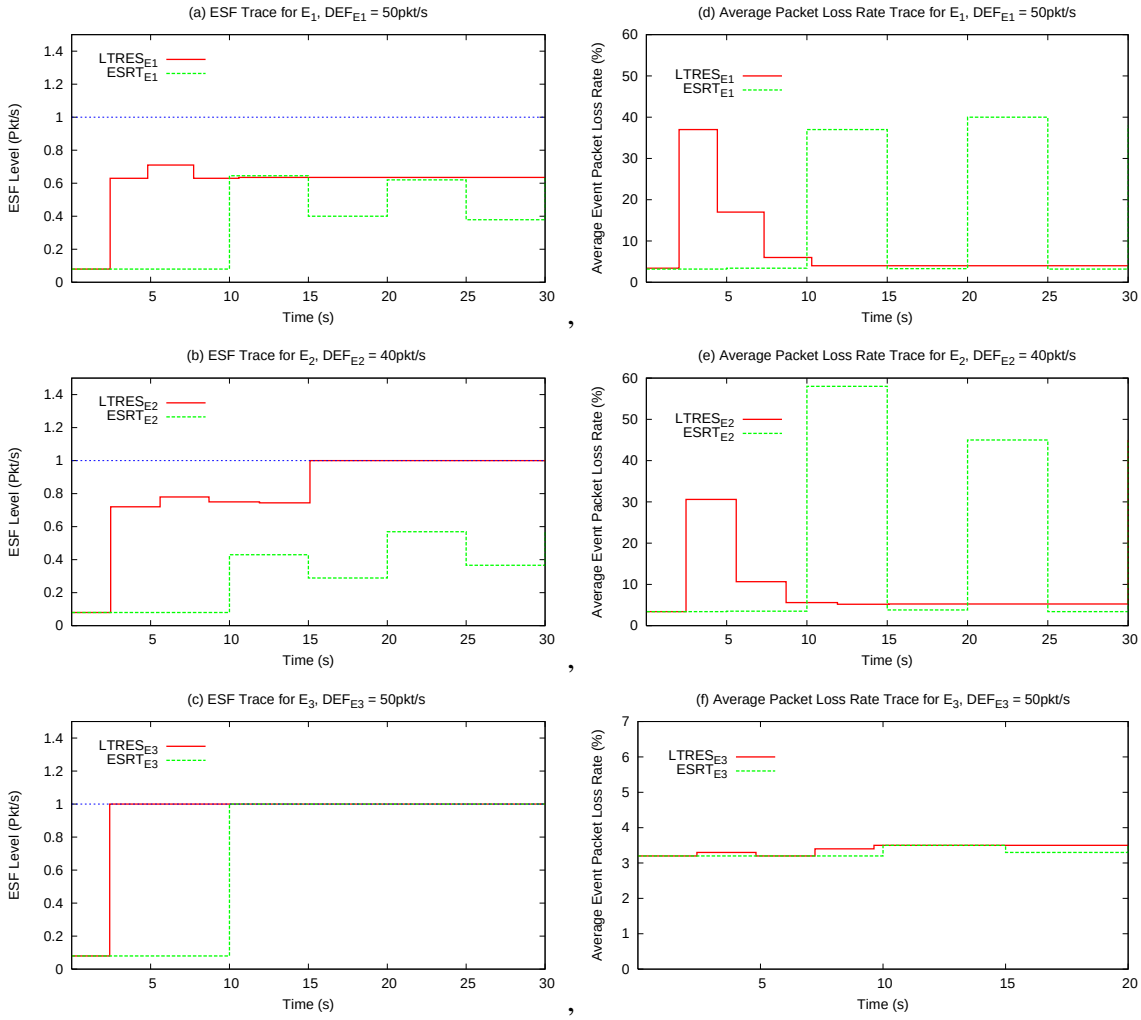


Figure 4.6: ESF level and average packet loss rate trace for LTRES and ESRT protocol in providing LTR services to three event areas, E_1, E_2 and E_3 ,where E_1 and E_2 compete for the channel in their end-to-end routes.

isolate the source rate adaptation operations at the events sharing the end-to-end routes so that the protocol operations can converge within limited time as for single event scenario. In order to do that, it requires the enroute nodes to check the 'Event ID' field in each bypassing packet to tell whether two events share the same node in the route. If so, ESRT will prioritize the source rate adaptation operations for these coupled events by their ESF levels so that ESRT can separately regulate the ESR at different events and converge in limited time. This isolation policy introduce much more control overhead especially in case of large number of events with high ESR. In addition, since ESRT cannot determine the unsustainable ESF requirement and discriminate the location congestion from the full congestion in its protocol operations, when the total number of event increases, the number of path-sharing events will also increase. As a result, the case of local congestion will increase in source rate adaptation process and the sustainable DEF requirement for each event will decrease, which makes ESRT only converges for very low DEF requirements. For the simulation scenario, as shown in Fig. 4.6, ESRT fail to converge for both events in multiple-event case although the DEF requirements are not higher than that in the single-event case.

4.5.3 Fairness Control on LTRES Flows

In this subsection, we show that how the steady-state throughput estimation algorithm help improve the fairness control on LTRES flows. Fig. 4.7 shows the average goodput distribution observed at the sink after LTRES operations in multiple event scenario. From the previous analysis, we know that LTRES provides LTR service to E_3 with only $Stage\ One$ operation. Since each Enode starts from the same r_d and performs the same MI operation, the fairness is guaranteed among the data flows originating from E_3. For E_1, based on the analysis in last subsection, all Enodes are detected to be congested in the first SAP. Therefore, all of them go through the Stage Two operations. From Fig. 4.7, we find out

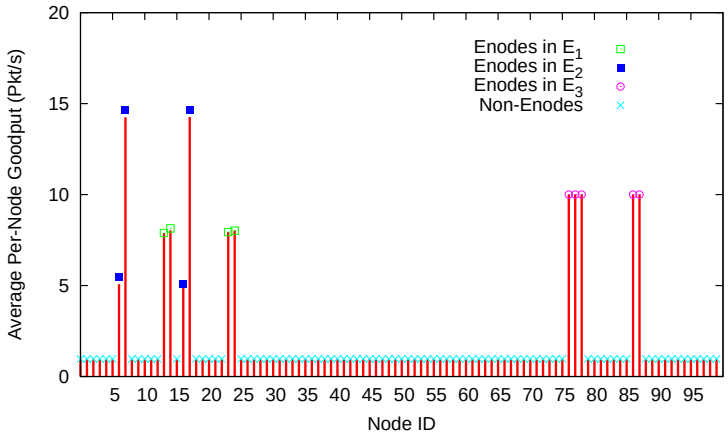


Figure 4.7: Average per-node goodput distribution after LTRES operation for multiple event scenario. The simulation results show that both $Stage\ One$ and $Stage\ Two$ operations result in a fair bandwidth allocation for LTRES flows sharing the congestion bottleneck.

that these Enodes share the same congestion bottleneck, thus similar goodputs are observed from these node. For E_2, the Enodes are divided into two groups $\{s_6, s_{16}\}$ and $\{s_7, s_{17}\}$. From Fig. 4.7, we find out that the sink gets similar goodputs from the Enodes within the same group. By revisiting Fig. 4.6 (b) and (e), this situation implies that only part of the nodes in E_2 shares the congestion bottleneck with E_1, while the rests keep active till the last SAP to provide the guaranteed LTR service. According to the above analysis, we conclude that both $Stage\ One$ and $Stage\ Two$ operations result in a fair bandwidth allocation in an event area for LTRES flows sharing the congestion bottleneck.

4.6 Summary

In this chapter, we propose LTRES, a distributed source rate control protocol, to provide LTR transport services for upstream data transmission in WSNs. LTRES can be applied to a continuous surveillance wireless sensor network with several event areas. Compared with earlier LSR data transport protocols, LTRES addresses fast and reliable event sensing with congestion control. Compared with an existing LTR data transport protocol ESRT,

LTRES provides both reliable data transport for sustainable LTR requirements and best-effort data transport services for unsustainable LTR requirements. It has faster convergence time, lower packet loss rate and better bandwidth utilization, especially for a high DEF level. LTRES also provides fair rate control for the distributed source rate adaptation. The future work on LTRES design includes implementing the protocol in a WSN testbed for performance evaluation under noisy and fading channel environment; considering the the Enode residue energy level in distributed rate adaptation mechanism to improve the network lifetime.

Chapter 5

A Real-time Groundwater Monitoring Network for Drought Assessment

5.1 Introduction

Groundwater resources are the principal source of drinking water for about 50 percent of the United States population. It also plays a major role in the Nation's agriculture [76]. However, groundwater resources are under increasing stress as there is a rapid growth in their usage. For instance, from year 2000 to 2007, large parts of Nebraska have experienced aquifer declines of one to five feet, as shown in Fig. 5.1. In some areas, irrigation pumping has even lowered groundwater levels by about 30 feet. Severe drought has been observed as a recurring problem and has raised serious concerns about our Nation's vulnerability to drought-induced water shortages.

Agricultural producers and other drought and water policy makers need more timely and accurate data to assess groundwater conditions to manage adverse situations such as drought and loss of pumpage in agriculture and domestic water supply [77]. For example, drought severity and duration indices are imprecise in detecting the onset, end, and

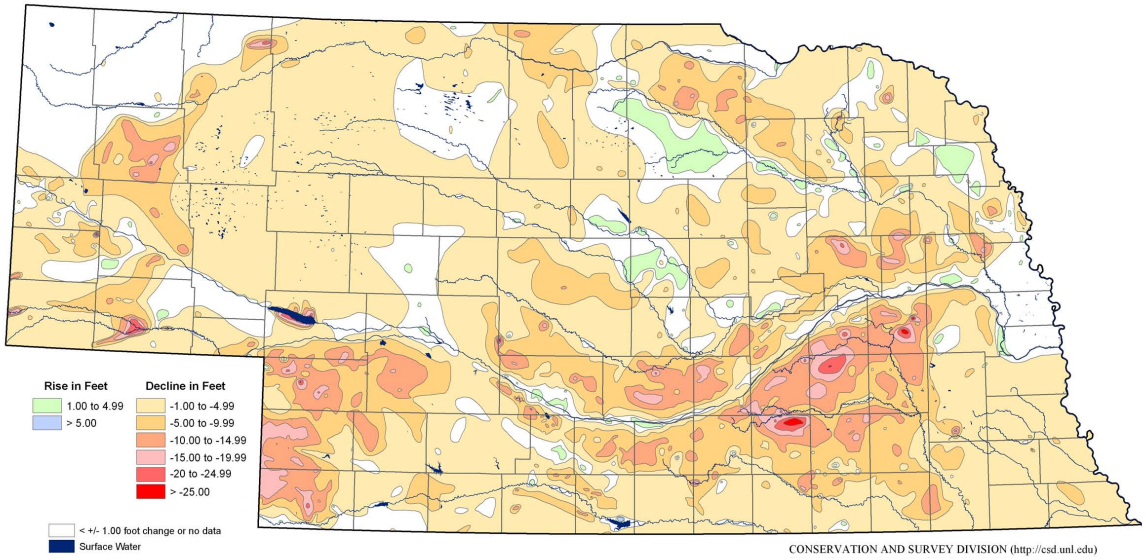


Figure 5.1: Groundwater level changes in Nebraska - Spring 2000 to Spring 2007.

accumulated stress of drought because they lack real-time groundwater level data [78]. At more than 20,000 ground water observation wells in Nebraska, groundwater levels are commonly collected once or twice a year using a hand-held tape to measure the depth. Some of the wells are equipped with stand-alone pressure transducers and recorded on an hourly to daily basis. However, a visit to the well is required to collect the data, which makes the existing groundwater level monitoring system labor intensive and not easily scalable if the number of logging locations increases. Although the groundwater level data are valuable to understand the resource usage, they are not available in a timely fashion or at the frequency that water resource managers need them to make relevant decisions [77].

The need for real-time groundwater level data suggests the development of large-scale, real-time groundwater monitoring infrastructures in drought-prone regions, such as Nebraska, for acquiring, transferring, analyzing and sharing groundwater level data. With the development of wireless communication and low-cost microprocessor technologies, instrumenting the distributed and remote wells with integrated sensing, processing and wireless communication units can make such an infrastructure feasible. The integration of local

sensing and processing allows the network administrators to apply data filtering, compressing and triggering functions. The wireless communication capabilities allow the integrated sensing unit to be re-programmed or re-tasked after deployment in the field. Therefore, each integrated unit has the ability to adapt its operation over time in response to changes in the environment, the condition of the sensor network, or various scientific endeavor.

To develop this large-scale real-time groundwater monitoring network by utilizing the advanced computing and networking technologies, we worked with environmental scientists in Nebraska. Our main goal is to apply the existing technologies in computer engineering area to help in improving the state-of-the-art in environmental monitoring. We seek to develop an effective monitoring network infrastructure for the domain, not just for a single groundwater monitoring application in Nebraska. Different environmental monitoring tasks can be carried on the same infrastructure; and the real-time data acquired by the network can be processed, analyzed, and accessed using the same platform to support various environmental research or natural resource management purposes. The project will also serve as a prototype for similar monitoring networks in other areas of the United States and internationally.

In this chapter, we discuss the design and implementation of a real-time environmental monitoring network in Nebraska that provides reliable groundwater level data with statewide coverage. The project, that is funded through a partnership with the U.S. Department of Agriculture Risk Management Agency, started in Dec,2005 and ended in Dec, 2009. Currently, reliable and real-time groundwater level data collected from 54 observation wells across the state are available on the World Wide Web for real-time public access or off-line data mining and analysis. Some early equipped sites have been operating unattended for more than 2 years without service interruption. Extension of the network with other types of sensors to provide versatile sensing data and establish comprehensive drought early-warning systems is expected in the near future. This network presents a col-

lection of requirements, constraints, and guidelines that serve as a basis for a general environmental monitoring network architecture for many such applications. We describe the core components of the network for this domain, including the architecture of the network, the communication method, the integrated sensing, processing and transmitting hardware installed at each well, and the data management facilities.

5.2 Background

5.2.1 Groundwater monitoring for drought assessment

Severe drought is a recurring problem for the United States and has raised serious concerns about our nation's vulnerability to drought-induced water shortages. *Droughts affect more people than any other natural hazard* [79] and result in serious economic, social, and environmental impacts. The Federal Emergency Management Agency (FEMA) estimates that droughts in the United States caused an average annual economic loss of 6-8 billion or 7.5-10 billion dollars in 2005. Estimates of the economic impact alone range from 10 billion to more than 20 billion dollars [80].

It is critical for drought-prone regions, such as the state of Nebraska, to establish comprehensive and integrated drought early-warning systems that incorporate climate, soil, and water supply factors such as precipitation, temperature, soil moisture, snowpack, reservoir and lake levels, streamflow, and ground-water levels [78]. Among these components, the ground-water level measurements from observation wells are the principal source of information about the effects of hydrologic stresses [81]. During times of severe drought, this hydrologic stress is most acute and can have an especially immediate and large impact on surficial groundwater [82]. However, *large-scale and real-time data is extremely limited for groundwater levels nationwide*.

Drought mitigation and monitoring programs have historically been reactive, empha-

sizing emergency response [78]. However, there have been serious limitations to efficient drought monitoring and mitigation programs. Some of the shortcomings of current drought early warning systems include [83, 78]:

- Inadequate density and data quality of hydrological networks and water-supply parameters;
- Inadequate sharing among government agencies and the high cost of data;
- Data and information products that are often not user friendly and users who are often not trained in the application of this information to decision-making;
- Inadequate indices for detecting the early onset and end of drought
- Data and information on emerging drought conditions that are often not delivered to users in a timely manner.

Groundwater monitoring is especially important for drought assessment in the areas across the western United States, where irrigation plays a major role in agriculture. A real-time and more accessible monitoring network would allow farmers, ranchers, and other decision-makers to make management decisions before significant groundwater shortages and related agricultural losses occur, as well as better understand aquifer trends for more long-term strategic planning.

Currently most wells in Nebraska are sampled every $6 \sim 12$ months, and the information is not readily available or comprehensible to most agricultural producers or other decision-makers. The National Drought Mitigation Center (NDMC) participated in a series of drought planning workshops across Nebraska in 2003. During these workshops, agricultural producers cited a need for a more accessible and real-time ground water monitoring network for better understanding local and regional aquifer trends, making short- and long-term management decisions, as wells as for documenting drought conditions.

In terms of short-term planning, real-time ground water data allows farmers to better understand aquifer fluctuations and match available water to crop demands, thereby creating a more sustainable system. Also, understanding aquifer trends during multiple-year droughts allows producers to make more informed decisions about what crops to plant, or if to plant crops, and whether new wells or water supply sources are needed. Similarly, ranchers could also find the monitoring network beneficial for making stocking and rotation decisions based on water availability. Finally, the network provides a tool to help producers document drought conditions for government assistance and insurance programs, such as prevented planting.

In terms of long-term planning, allowing producers to analyze local historical groundwater-levels helps them better understanding basic trends and sustainable withdrawal limits. It may also help them better realizing the long-term viability of ground-water expectations in their region. As our water resources are increasingly utilized by a wide variety of users, the need for monitoring data becomes even more important to more effectively manage our limited ground water resources.

At the national level, the NDMC is also a partner in developing the US Drought Monitor, a weekly assessment of current drought conditions around the United States put together by a collaborative effort between the NDMC, three federal agencies, and 160 other state and federal representatives who provide local drought information. Our groundwater monitoring network will serve as a prototype for similar monitoring networks required by NDMC and other collaborators to better assess hydrologic conditions in the state. The real-time groundwater level and other hydrologic data collected by the network can then be incorporated into the US Drought Monitor, which has become a common source of drought-related information for state and national decision-makers and is used extensively by USDA and others in making agricultural policy decisions.

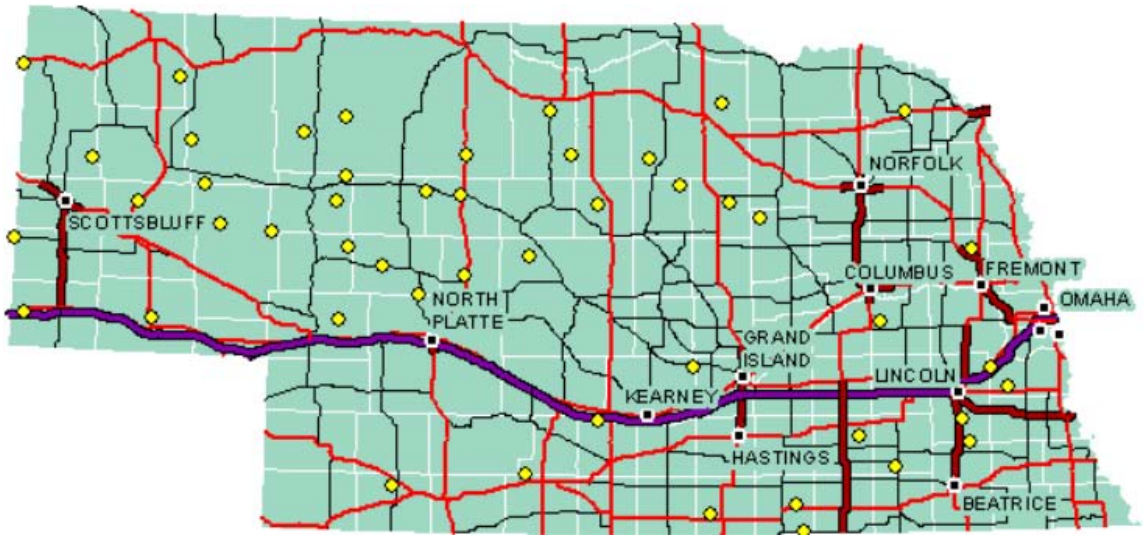


Figure 5.2: The site map of the state-wide real-time groundwater monitoring network.

5.2.2 Requirements for Groundwater Monitoring in Nebraska

The developed groundwater monitoring network meets the following requirements and constraints, which also outline the basic criteria for a general environmental monitoring network for similar applications.

Wide and wild area coverage

The network contains more than 50 monitoring sites to cover the entire state of Nebraska with area of 200,520 square miles, as shown in Fig. 5.2. The linear distance between any one site and its closest neighbor is about 20-100 miles. Monitoring sites were primarily chosen for their ability to detect the onset, magnitude, and recovery of hydrological drought. Sites are usually located in rural areas to minimize the pumping effect on the aquifer, where land line or cellular services may not be available. Observation wells located at these sites have to be shallow to water (< 100 ft to water), permeable to the overburden, open to a single hydrological unit, and a horizontal extent of the hydrologic unit the well penetrates.

Scalable network and flexible sensing tasks

The first stage of network deployment requires the groundwater level of $1 \sim 4$ observation wells to be monitored in the surrounding area of each selected site. However, to establish a comprehensive and integrated drought early-warning system, more sensors are expected to be installed in the surrounding area to conduct precipitation, temperature, soil moisture, snowpack, water levels, and stream-flow monitoring. The expending network scale and flexible sensing tasks require a hierarchical wireless network architecture. At each monitoring site, a wireless local sensing network is required to provide the short-range connectivity among the large number of sensing units covering the surrounding area. The data collected by these sensing units can be forwarded in the local wireless network to a local processing and transmitting center. New sensing units equipped with low-cost and low-power transceivers can be easily added into the local wireless network without changing the network structure. A relatively high-cost backbone wireless network then needs to be constructed to provide the reliable long-haul communication between the local network and a central base-station located at the University of Nebraska-Lincoln.

Long-term unattended operation

The expected service-life for the real-time groundwater monitoring network is 10 years. The integrated sensing, processing and transmitting units are installed by project personnel at each observation well included in the network. After proper tuning and configuration, the hardware and software installed at the well will be left unattended for the rest of service-life, unless severe hardware or software failures occur. Therefore, the hardware and software system should be robust and resilient to severe environmental effects. Renewable energy is required at these sites to power the system. In our network, solar power is used as the energy source. The insolation duration in Nebraska guarantees sufficient solar power to drive the stand-alone equipment with low probabilities of service interruptions due to power loss.

Reliable and real-time communication

The groundwater level data collected by the sensing unit are processed and transmitted as packets to the base-station in an hourly manner. The communication channel between the observation wells and the base-station should be reliable to guarantee high data packet receiving rate. Proper data redundancy is required in data transmission so that occasionally lost data packets can be recovered.

Management at-a-distance

The remoteness and wide-coverage of the field sites require the ability to monitor and manage networks infrastructure at the base-station. After the hardware and software systems are installed at the remote site, the network manager is able to monitor the operation status of the equipment at each well, such as signal strength, power level, data rate, and etc., by on-line tools. The network manager should also be able to re-task the observation node remotely, including tuning the data sensing frequency, modifying the data processing or filtering algorithm, and etc.

Fast and easy data access

The data collected by the network should be reactively and instantly decoded and stored at the base-station for public access or off-line data mining and analysis. Consistency and error checks are required before the data can be store. The hydrological data should be presented and available in a variety of formats to aid decision-makers and producers. An online access site is important to allow visitors to view current conditions in hydrography or tabular form and to choose a temporal range. End users may also choose to incorporate the data into their own GIS presentations.

5.2.3 Enabling Technologies

In this subsection, we give a brief overview on the enabling technologies for real-time environmental monitoring, with a focus on the wireless communication method.

Wireless Sensor Networks

Wireless Sensor Networks (WSNs) has been emerging as a promising technology for observing the physical world with low-cost and high accuracy. During the past few year, a number of real-world WSNs have be deployed for environmental monitoring related applications, such as the Great Duck Island habitat monitoring network [2], the ecology monitoring network in Hawaii [84], the Columbia river coastal margin observation and prediction network, CORIE [85], the volcano eruption monitoring network in central Ecuador [3], the animal behavior and environmental interactions monitoring network in northern Australia [86], and the James San Jacinto Mountains Reserve forest monitoring network in California [87].

WSNs usually use WiFi (IEEE 802.11) [88] or ZigBee (IEEE 802.15.4) [89] compliance wireless communication technologies with low-power radio design. ZigBee is targeted at radio-frequency applications that require a low data rate and long battery life. The ZigBee transceivers usually have very low transmission power, ranging from $0dBm \sim 3dBm$, and receiver sensitivity, ranging from $-85dBm \sim -100dBm$. The maximum transmission range of ZigBee devices is usually $20m \sim 50m$ for indoor communication and $80m \sim 100m$ for outdoor communication. The maximum data rate of ZigBee device can reach $250kbps$.

WiFi is targeted at radio-frequency applications that require higher data rate, secure networking and mobile communication. According to the latest IEEE802.11g standard, the maximum data rate of WiFi device can reach $54Mbps$. The WiFi transceivers usually have higher transmission power, ranging from $10dBm \sim 20dBm$, and varying receiver

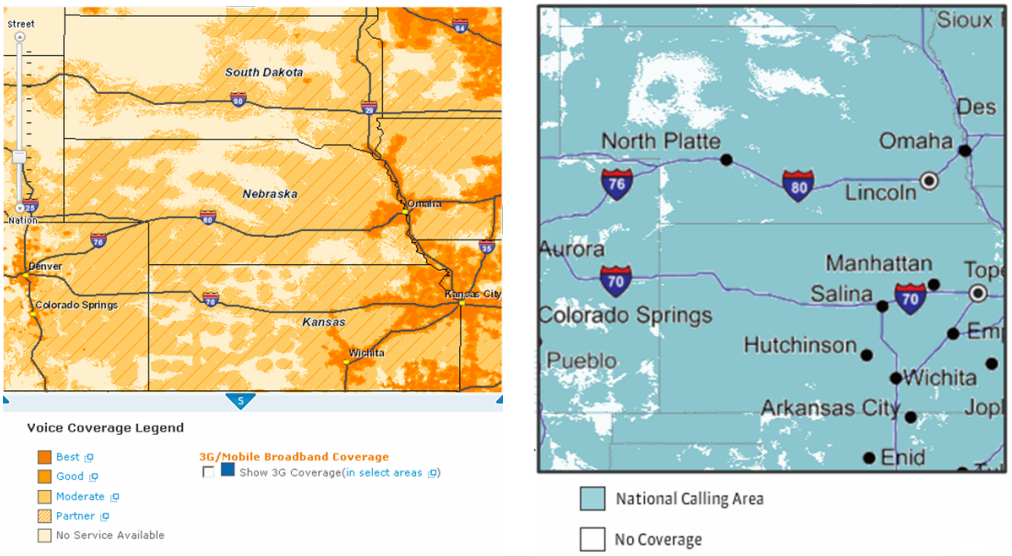
sensitivity, ranging from $-70dBm \sim -90dBm$, depending on the required data rate. The maximum transmission range of 802.11g devices is usually $45m - 60m$ for indoor communication and $75m \sim 100m$ for outdoor communication. By adding multiple-input multiple-output (MIMO) antenna design, the transmission can reach $70m$ for indoor communication and $250m$ for outdoor communication.

WiFi and ZigBee technologies work best for local environmental monitoring network with flexible sensing tasks. Sensing units integrated with low-power radios form the multi-hop WSN. Since WiFi and ZigBee technologies feature low-cost design, ($\$50 \sim \$200$ per node), large number of sensor node can be deployed in the local monitoring area to improve the sensing resolution. Different types of sensors can be deployed in the adjacent area to extend the sensing scale. The sensor-to-sensor communication capability enables complicated data integration and event detection algorithm to be developed for comprehensive monitoring tasks. High network density also help to increase the fault-tolerance and robustness of the system.

However, relatively short transmission range and high data loss rate limits the scale of WSN deployment for environmental monitoring. For example, to cover a $1km^2$ 2-D sensing area, about 250 radios are required to provide the minimum connectivity using WiFi or ZigBee technologies. Significant signal attenuation introduced by obstacle and fluctuate terrain in the real-world deployment can greatly reduce the achievable transmission range and packet receiving rate. Therefore, topology control and reliable communication protocol, such as those proposed in this dissertation, are necessary for WSN to achieve the reliability and timeliness requirements in environmental monitoring applications.

Cellular or Land Line Phone Networks

Cellular or land line phone networks have been used widely to provide the duplex wireless communication in the traditional large-scale environmental monitoring applications,



(a) AT&T (b) US Cellular

Figure 5.3: Limited cellular coverage on rural areas in Nebraska.

where cellular or telephone modems are used to modulate and switch the data collected from the remote site network into the phone network and demodulates the information from the carrier network to the base station. A phone network terminal, including the modem, processor and antenna, usually costs $\$900 - \$1,800$. Depending on the data plan purchased from the service providers, such as AT&T and US Cellular, the sensing data can be transmitted at a certain data rate over the phone network, usually range from 5 to $50kbps$. Limited bandwidth is also available for compressed sound or video transmission. Dial-up is the common switch-in service provided in phone network communication. Short Message Service (SMS) services are also available from some service providers for data communication. In this case, all the sensing data switched into the network will be forwarded to a message service center, and then can be downloaded through Internet.

Coverage is the main problem for using cellular or land line phone networks in many rural environmental monitoring applications. As shown in Fig. 5.3, most phone network service providers only have good signal strength around major cities. Expensive directional

antennas are required in low coverage or hilly areas to boost the signal and guarantee the transmission reliability. Recurring data service fee also need to be considered while using phone network for backbone network communication in a long-term monitoring application. A $\$20 - \$40$ data connection fee is required for each service line to transmit up to $200MB$ data per month.

Satellite Communication Networks

Satellite communication systems have been used to provide the long-haul wireless communication in large-scale environmental monitoring applications. A number of competing satellite communications systems are currently available with full coverage on all area of North America, including GOES, Iridium, Inmarsat, ARGOS, and Globalstar. Satellite transceivers are used at the remote monitoring sites to modulate and switch the sensing data collected from the local network on to the the satellite communication channel. A satellite terminal, including the transceiver, antenna and processor, usually costs $\$2500 - \$4000$. Depend on the available data communication services, the transmitted date can be received over Internet through PPP service, Short-Burst-Data service and SMS service, or by another Satellite transceivers located at the base station through Dial-Up service. [90] Compared with cellular or land line network, satellite communication is more reliable in term of signal quality and coverage. At some rural environmental monitoring sites, satellite channels currently remain the only solution to provide reliable wireless connectivity.

Limited bandwidth and relatively large delivery latency are two major problems faced by satellite communication for environmental monitoring applications. As shown in Table 5.1, most satellite communication services can only provide data rate less than $500bps$. The delivery latency ranges from $20s$ to 2 hours. For some real-time monitoring applications with large raw data sets, such as to transmit voice/video streams or bulk data collected from a local WSN, satellite communication may not be able to meet the minimum throughput

Table 5.1: Comparison Among Message Based Satellite Communication Systems

System	Message Size	Monthly Cost(msg/hr)	Anytime Cost
Iridium	$< 340B$	\$32	$\$13/mo + \$0.0015/B$
ARGOS	$32B$	\$437	$\$21/mo + \$3.50/6hr$
GOES	$400B$	$Free$	$Free$
Inmarsat	$25B$	\$90	\$0.06 for $10B$
Globalstar	$36B$	\$165	$\$30/mo$ for $100msg$

System	Peak Power Cons.	Two-Way?	Data Rate	Delivery Time
Iridium	$1.8W$	Yes	$2.4kbps$	$< 20s$
ARGOS	$1W$	Yes	$480bps$	$< 2hrs$
GOES	$50W$	No	$100/300bps$	$< 1hr$
Inmarsat	$9W$	Yes	$480bps$	$30s$
Globalstar	$5W$	No	$100bps$	$< 30mins$

and latency requirement. The limited packet size also introduces higher control overhead and reduce the achievable goodput. High peak energy consumption is another problem faced by many satellite communication systems. Compared with phone network communication systems, that only draws $500mw \sim 800mw$ peak power in transmission, satellite communication systems usually consume $5 \sim 10$ times more power, thereby requires more expensive and bulky power system. High equipment cost and monthly service fee are other factors that need to be considered while using satellite communication channel for backbone network communication.

5.3 Design and Implementation

5.3.1 Network Architecture

According to the application requirements described in Section 5.2.2 and the enabling technology discussed in Section 5.2.3, we developed a two-tier architecture for the state-wide real-time ground monitoring network. The full architecture is depicted in Figure 5.4.

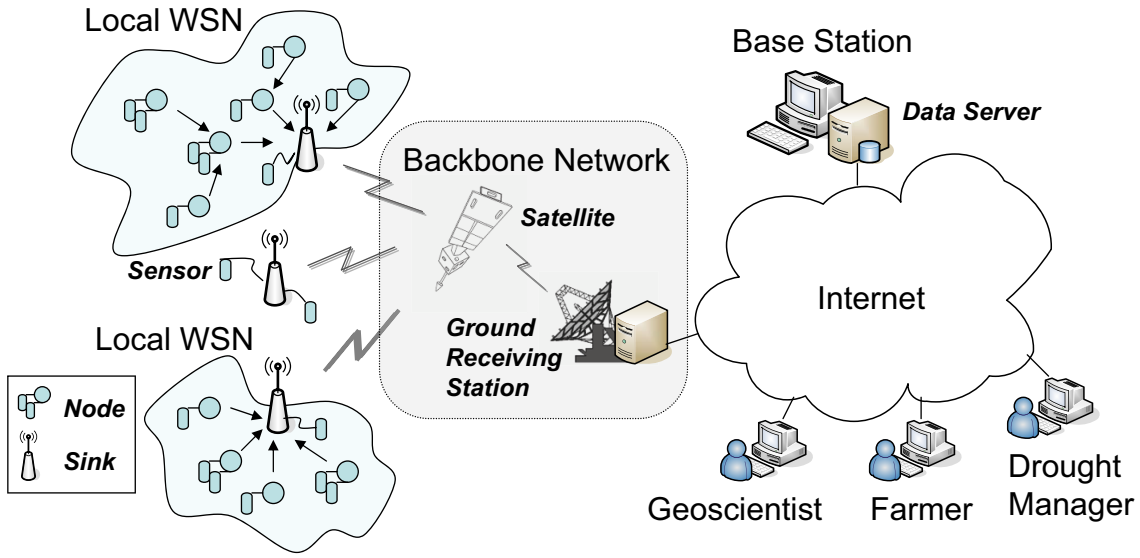


Figure 5.4: Two-tier network architecture for a state-wide real-time ground monitoring network.

The tier-one network is a wireless backbone network that provides the reliable long-haul connectivity between each monitoring site and our network base-station. Since the sensing tasks conducted at each site are usually independent and the communication between two sites is rare, a star topology is used in the tier-one network. The main responsibility of the backbone network is to relay the sensing data collected from the local monitoring site to the base station. A network re-tasking command may also be sent through the backbone network to the selected monitoring site to reconfigure the on-going sensing tasks, such as changing the sample frequency. Because of the wide and wild coverage required by the groundwater monitoring application, we choose satellite communication systems in the tier-one network. A satellite transceiver is required at each monitoring site to communicate with a geosynchronous orbiting satellite. The sensing data can be relayed to a ground receiving station through the satellite during selected time windows or short burst-data service, depending on the particular monitoring tasks. The base station located at the university communicates with the satellite ground receiving station through Internet to retrieve the sensing data collected from all monitoring sites, and store the pre-process data

in the database. The end users, including geoscientists, farmers, and drought managers can access the real-time or historical sensing data through on-line interface in tabular and graphic forms or to incorporate the data into their own GIS presentations.

The tier-two networks are WSNs deployed at selected monitoring sites to extend the sensing scale in these areas. Each tier-two network has one satellite communication center serving as the sink. A number of sensors conducting different sensing tasks are deployed in the surrounding area to collect various hydrological data so that an integrated drought model can be derived based not only on groundwater level information but also on the climate, soil and water supply factors. Depending on the locations of the sensors, one or more sensors may be connected to a processing unit integrated with a WiFi or ZigBee transceiver. We call this integrated processing and communication unit that connects with one or several sensors, a node, in a tier-two network. Multi-hop communication is enabled among the nodes and the sink. A mesh, ring or star network topology is adopted, depending on the coverage of the network and the distance between the node and the sink. Various sensing data collected by the tier-two network can be cached at the node, or forwarded to the sink. Local data aggregation or event detection algorithms can be applied in tier-two networks. The network re-tasking command can also be disseminated from the sink through broadcast.

The two-tiered network architecture not only enables easy network extension at particularly interesting sites with dedicated sensing tasks, but also helps minimize the network maintenance cost. According to the cost information provided in Section 5.2.3, the tier-one network is reliable but expensive; while the tier-two network is cost-efficient but prone-to-failures. Therefore, in our network architecture, each monitoring site has one tier-one network access point to guarantee the reliable communication between the site and the base station. The disconnection or device failure in one tier-two network will not affect the monitoring tasks conducted by the rest of the network. It also makes the failure point de-

tection easier in a large-scale monitoring network. A local network failure can be quickly located and the diagnosis can be conducted remotely through trusted tier-one connection. Project personnel can also be sent to a specific tier-two area for maintenance purposes. In addition, tier-two network nodes are low-power WiFi or ZigBee devices communicating over uncommercial channels. No communication service fee is required within tier-two network. Therefore, adding new sensing spots to the surrounding area of an interesting site only requires the installation of one or few nodes to provide wireless connectivity, which provides great flexibility for the long-term monitoring task with minimized cost.

5.3.2 Implementation and Deployment

Tier-One Communication System

Based on the proposed network architecture and application requirements, the tier-one network must be able to provide highly reliable communication for monitoring sites located in rural areas. we decide to use satellite communication system for tier-one network. The cellular or land line phone network cannot be used in our network implementation because most of our monitoring sites located in rural areas where phone networks provide very limited or no connectivity. Other long-range radio communication techniques, such as WiMax (IEEE 802.16) [91], also are not considered because the multi-hop communication required by these technologies for a state-wide coverage cannot guarantee the network robustness and reliability.

Among all the available satellite communication systems with full coverage in Nebraska, we choose Geostationary Operational Environmental Satellites (GOES) system [92] for our network implementation because it provides free services to government and no-profit users. In our network, each monitoring site is located more than 20 miles away from its nearest neighbor, which means each monitoring site requires a dedicated satellite

communication access point to transmit the data to the base station. According to the information listed in Table 5.1, using GOES system help us bring down the annually service fee by about $\$50,000$ for the more than 50 monitoring sites required in the network. However, GOES system currently only support one-way communication mode, which means the data can be transmitted to the base station but the command cannot be transmitted to the sites. A two-way GOES communication system is under testing and will be released in 2010 [93].

We use Sutron Satlink2 [94] to provide GOES access point for each monitoring site. A Satlink2 consists of four main components: satellite transceiver, data processor, memory and General Purpose Input/Output (GPIO) interface. The data input/ouput interface has 12 digital and 4 analogy ports, which allow up to 16 sensors to be connected to the GOES access point. The data input/ouput interface also includes a DB-9 RS-232 serial port, which allows PC or PDA devices to configure the GOES access point through cable connection or remotely through serial-bluetooth converter. The DB-9 serial port can also be used to connect a tier-two network node, which works as the sink for the tier-two network. The input sensing data can be preprocessed or filtered by the data processor and packaged for transmission. The preprocessed data can also be cached on a local circular flash memory with 1M capacity, which provides temporary data storage for data recovery in case of communication failure. A Sutron 5000-0080 YAGI Satellite Antenna is connected to Satlink2, providing enough beamwidth to illuminate at least two of the GOES satellites for reliable communication.

According to the application requirement, the GOES access points are set to work under hourly self-timed mode, which means the sensing data collected from the site are transmitted hourly in user assigned time windows. (Other kinds of services, such as random or interrogated connections, are also available through GOES system [95].) A Global Positioning System (GPS) receiver is equipped on each GOES access point to obtain accurate time synchronization. A GOES access point is only activated during the user assigned time-

window, while remaining in standby mode for the rest of the time. Within each assigned time-window, a GOES access point first need to synchronize with the satellite system by listening to a beacon broadcast by the satellite. It then ties to register with the network by sending a service request message containing a unique 8-bit GOES address. The service request message needs to be forwarded by the satellites until it reaches the GOES ground control station located at Wallops Island, VA. If the service request is authorized, a confirm message will be sent back to the GOES access point. After that, the GOES access point can send the data in limited time slots (usually less then 90s) at 100 or 300bps. The raw data packages will be forwarded by one or more satellites to the GOES ground receiving station and can be accessed by the authorized GOES users through Internet.

Tier-Two Communication System

According to the proposed network architecture and application requirements, the Tier-Two networks provide low-cost optional extension to the tier one network at selected monitoring sites. Commercial WSN devices are used in tier-two network implementation.

We use the IRIS mote from Crossbow Technology [96] as the nodes in tier-two networks. IRIS mote uses ZigBee (IEEE 802.15.4) compliant RF transceiver to provide bidirectional communication at 2.4 to 2.48 GHz band. The maximum data rate is 250kbps. It uses an Atmel ATmega1281 low-power micro-controller running at 16MHz, a 8kB RAM, and a 512kB serial Flash to store up to 100,000 measurements. The IRIS standard 51-pin expansion connector supports up to 10 analog inputs, digital I/O, I2C, and serial ports. These interfaces make it easy to connect to a wide variety of sensors. Each IRIS mote has preloaded open-source network protocol stack and operating system, MoteWorks^{TM} [97], in its internal flash memory, which provides ad-hoc mesh networking and over-the-air-programming capabilities. Users can modify or overwrite the default on-board protocol/OS, and add their own processing application based on the specific requirements.

Compared with other commercial ZigBee wireless modules, IRIS features 500 meters outdoor line-of-sight radio range by a high receive sensitivity setting, which is more than three times improvement over average. It uses direct sequence spread spectrum radio, which is resistant to RF interference and provides inherent data security.

We use MIB510 gateway board from Crossbow Technology [98] to connect the local tier-two WSN with tier-one satellite network. A selected IRIS node that serves as the sink of the local WSN, can be connected to an MIB510 gateway board through the standard 51-pin expansion connector. The MIB510 gateway board then connects the WSN sink to the GOES access point, Sutron Satlink2, through the RS-232 (DB-9) interface. By this way, the data collected from the tier-two network can be read, cached and packaged by Satlink2 for transmission during assigned time-windows. The local WSN reconfiguration command, such as changing the sampling frequency or changing the network routing topology, can also be sent from a PC or PDA device connected with the Satlink2, to the sink IRIS mote, and disseminated in the network using default MoteWorks or customized protocol stack. In the near future, with the release of two-way GOES communication system, full network re-tasking can be implemented by sending various control commands from the base statio to any tier-two WSN according to the changing sensing applications and requirements.

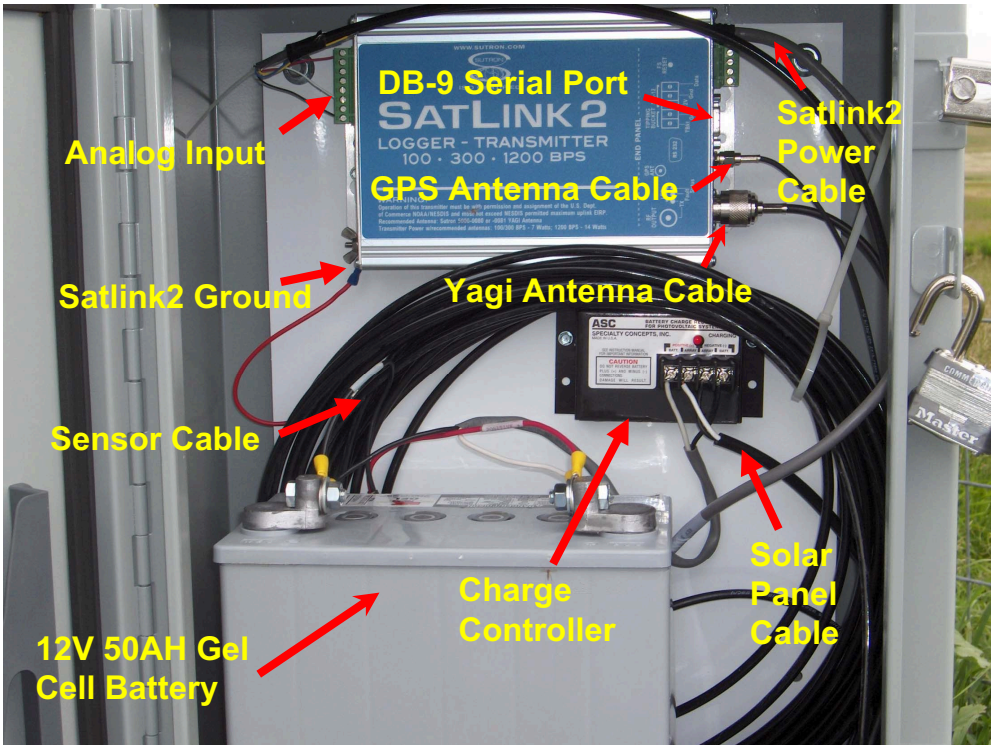
On-Site Deployment

We started the feasibility research and the prototype testing for the proposed network architecture and implementation strategies in May 2006. The deployment of the state-wide real-time groundwater monitoring network started from March 2007. The deployment and testing of tier-one network finished in December 2009, which successfully provides reliable wireless connectivity to 52 monitoring sites covering the entire state of Nebraska [99]. A complete site map is shown in Fig. 5.2. Currently, each GOES access point wirely is connected in a wired network to 1-4 pressure transducers for monitoring the groundwater level

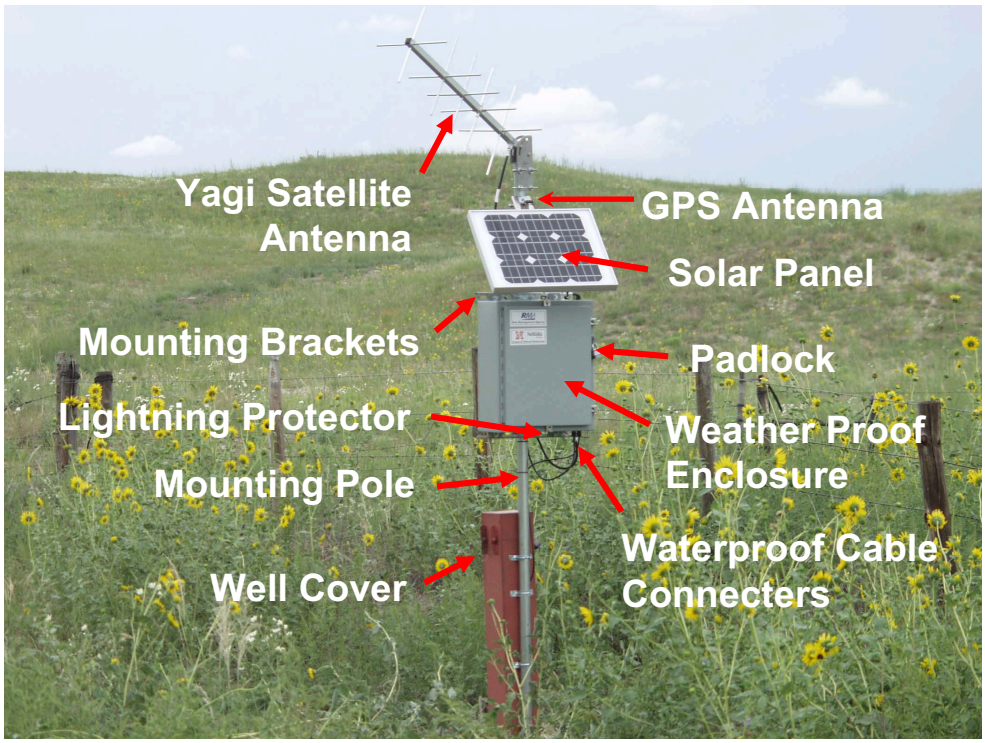
of surrounding observation wells. The deployment of tier-two WSN at selected monitoring sites to construct comprehensive and integrated drought early-warning system with precipitation, temperature, soil moisture, snowpack, water levels, and stream-flow monitoring is under planning.

To deploy the proposed system for remote outdoor monitoring applications, how to provide reliable and sufficient power to the stand-alone equipment, and how to protect the system under severe weather conditions are two vital problems. As we mentioned in Section 5.2.2, the expected service-time of the network is 10 years. Therefore, renewable energy source is required at each monitoring site to power-up the system. For a GOES access point, Satlink2 is the major power consumer. A Satlink2 draws 3.8A in transmitting status and 6mA in standby status (including interrogating the IRIS sink or directly connected sensors). The power drawn by IRIS node and the pressure transducer connected to the Satlink2 is negligible. If we assume the Satlink2 keeps active during a 30s time-window for completing the hourly registration and transmitting, and keeps standby in the rest of the time, a GOES access point draws about 1.2AH per day.

In our deployment, solar power is used as the energy source to power-up the system. The solar power system located at each GOES access point consists of three main components: solar panel, charge controller and battery. We use 12V 50AH sealed Gel cell battery from MK Battery's Co. for the deep cycle service, which means stable long-term energy delivery instead of burst energy delivery, required by our application. The capacity of 50AH can power-up the GOES access point for about one month without recharge. Gel cell battery excels in slow discharge rates and offer greater reliability in extreme temperature applications. It also provide maintenance free energy delivery with leak/spill proof. One big issue with Gel batteries that must be addressing is the charge profile. Gel cell batteries must be recharged correctly or the battery will suffer from premature failure. The battery charger being used to recharge the battery must be designed or adjustable for Gel



(a) Inside the weather proof enclosure



(b) External view

Figure 5.5: The internal and external view of the an on-site GOES access point deployment.

Table 5.2: A List of Equipments Used in the Network Deployment

Equipment Details	**Provider**	**Unit Price**
SatLink2 Satellite Transmitter and Logger	Sutron Corp.	$2,367.00
5000-0080 YAGI Satellite Antenna with 15 ft Cable and Mounting Bracket	Sutron Corp.	$344.00
Lightening Protector, with COAX Cable	Sutron Corp.	$65.00
BSP2012 Solar Panel, with 18ft Cable and Mounting Bracket	Power-Up Corp.	$241.00
ASC Solar Panel Charger Controller, 4A	Specialty Concepts INC.	$44.00
Sealed Gel Cell Battery, 12V 32AH	MK Battery	$104.00
Acculevel 15 PSI Submersible Transducer, with 50ft Cable	Keller America	$528.00
Hoffman A20H16 Stainless Steel NEMA 4X Enclosure with Back Plate	Crescent Electric Supply	$258.00

cell batteries, with typical absorption voltage ranging from 14.0 to 14.2 volt and float voltage ranging from 13.1 to 13.3 volts. We use the ASC photovoltaic battery charge controller from Specialty Concepts INC. to provide the strictly controlled low voltage recharge for Gel cell batteries. ASC has a simple, low component count design that is completely solid state and sealed against harsh environments. A blocking diode is also included in ASC charge controller for preventing battery discharge during low or no light conditions. To recharge the battery with enough solar power, we use BSP 2012 solar panel from Power Up Solar, to provide 20W maximum power. It can fully charge the 50AH battery from empty in about 37.5 hours under direct sunlight (15W delivers about 1 amp per hour). According to the average sun hours per day in Nebraska, the solar panel can fully recharge the battery in about 8 days.

To withstand the variable weather conditions in harsh outdoor environment, we use weather-proof Hoffman A20H16, 20" X 16" X 10" Stainless Steel NEMA 4X Enclosure to encapsule the electronic and power system. The internal views of the enclosure is given in Fig. 5.5 (a). The external components at a GOES access point is shown in Fig. 5.5 (b). At

each GOES access point, the weather-proof enclosure is mounted on a vertical mounting pole. The Yagi satellite antenna and the solar panel are mounted at the top of the pole with properly tuned angles. All the cable via holes on the enclosure are secured with waterproof connector. Table 5.2 shows a list of equipments used in the network deployment with their provider and pricing information.

Base Station

The base station of the network is located at University of Nebraska - Lincoln. The sensing data and other performance parameters collected from each monitoring site are transmitted through the GOES assess point and forwarded by the GOES satellites to the NOAA Local Readout Ground Station (LRGS). From there, the data can be retrieved by the authorized GOES users through the Internet using Data collection platform Data Service (DDS) protocol [100]. Currently, four types of services, NetList management service, network status monitoring service, data management service and web service, can be provided by the base stations, as shown in Fig. 5.6.

The NetList management service and network status monitoring service are provided based on NetList database, which contains the information about each monitoring site within the network, including the site ID, the site location, the installed sensors and other data communication and data processing parameters associated with the site. The network managers can add/remove a monitoring site or change the parameters of a site through the NetList management service. The network managers can also monitor the status of a site in the NetList database through the network status monitoring service, which can draw the current and historical signal strength, battery voltage, transmission frequency offset, and other performance parameters of the site from the NOAA LRGS data server. The performance parameters can help the network managers to remotely diagnose a node failure at the base station. In our current implementation, the NetList management service and net-

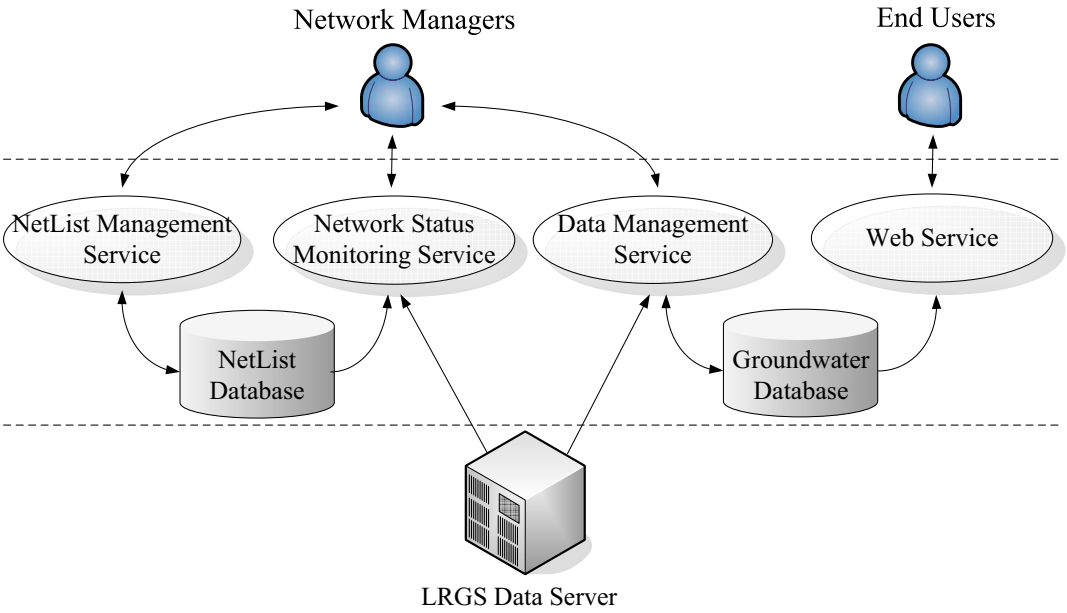


Figure 5.6: The network and data management services provided at the base station.

work status monitoring service are developed based on an open source LRGS client-side software suite DECODES [101].

The data management service is responsible for automatic data retrieving from the NOAA LRGS data server according to the site information stored in NetList database. The retrieved raw data are then decoded and processed to desired engineering formats according to the specific application requirements. After proper data consistency and correctness check, the time-tagged sensing data collected from each monitoring site are inserted into the groundwater database at the base station. A complete data communication and processing flow chart of data management service is shown in Fig. 5.7. The network manager can also modify the data decoding and processing algorithms or alter the data stored in groundwater database through an user interface provided by data management service. A web server is connected to the groundwater database to provide various web services for the end users. The web service visitors can browse the monitoring network through a 2-D site map, view current or historical groundwater conditions of interesting areas in hydrography or tabular

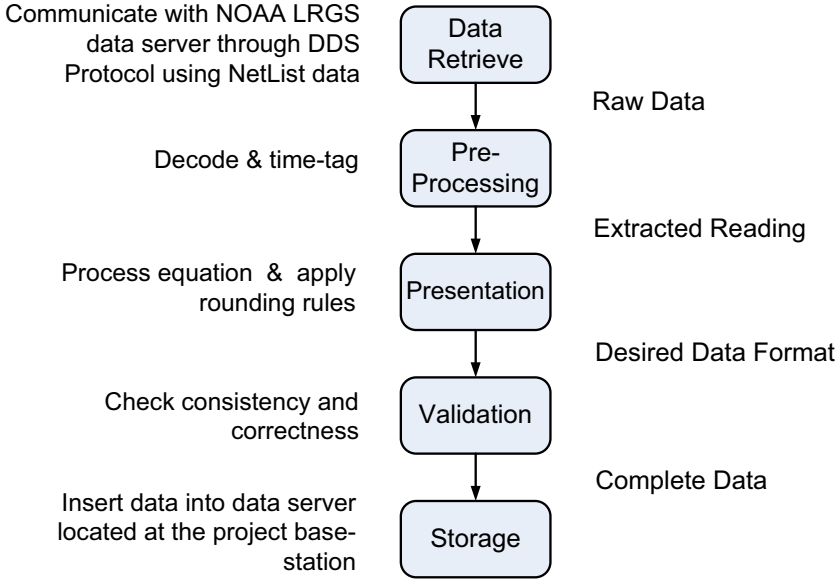


Figure 5.7: Data communication and processing flow chart of data management service provided at the base station.

form. End users may also choose to correlate the groundwater level data with temperature/ precipitation data collected in the surrounding area or download and incorporate the groundwater level data into their own GIS presentations.

5.4 Current Results and Discussions

A real-time groundwater monitoring network with 50 major monitoring sites are deployed in Nebraska, of which 4 are equipped with 1-4 sensors. Currently, the tier-one network has been fully deployed and tested to provide reliable long-haul communication between sites and base station. The deployment of the tier-two network at selected monitoring sites is under planning. The designed solar power system has been tested to provide enough power for the GOES access point under various weather conditions. The earliest deployed site has been conducting the continuous unattended monitoring tasks for two and half years. Only a few service interruptions have been observed at totally 4 wells due to equipment failure or

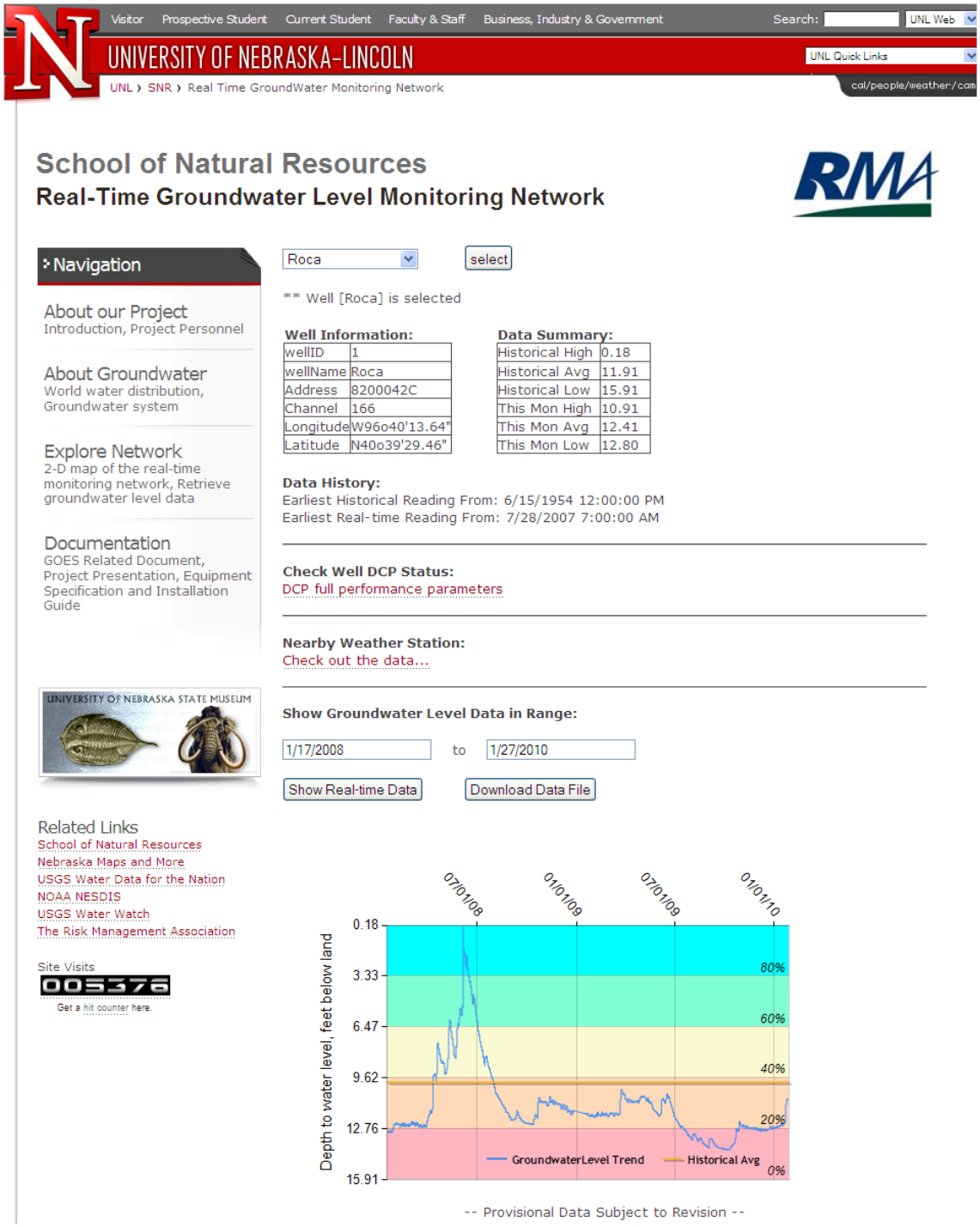


Figure 5.8: A snapshot of the groundwater monitoring network web interface.

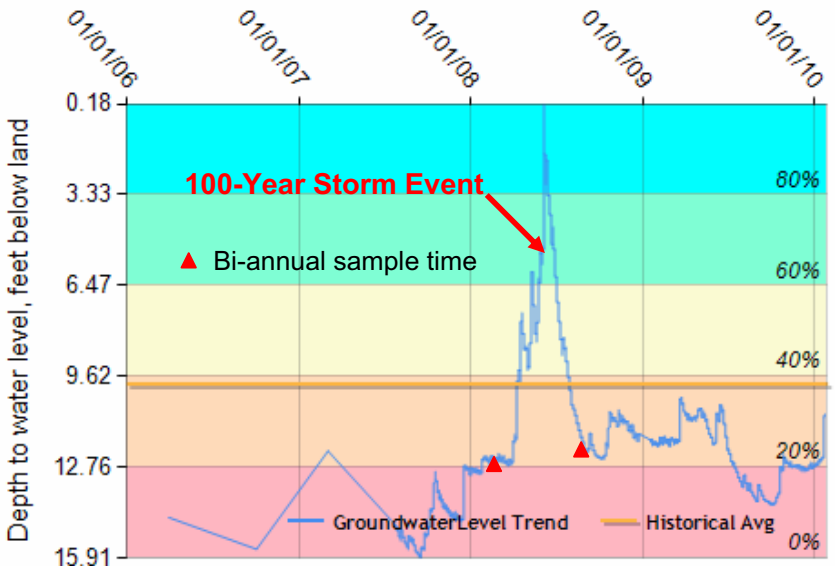


Figure 5.9: The groundwater level data collected from an observation well located near Roca, NE, show the groundwater fluctuation during a 100-year storm event.

heavy icy rains temporarily changing the antenna parameter. The groundwater data users can visit out web server (snr-1349.unl.edu) to browse the site map, view the real-time or historical groundwater level data from an observation well, or download and incorporate the selected data into their own GIS presentations. A snapshot of the on-line data retrieving page is shown in Fig. 5.8. The groundwater level data collected by the network have already been used by geoscientists in Nebraska to create the latest state groundwater map and will be integrated with National Integrated Drought Information System (NIDIS) to derive support decisions.

Fig. 5.9 shows the groundwater level curve of an observation well near Roca, Nebraska, given by our web interface. The data from July 28, 2007 to January 1, 2010 are collected by our real-time groundwater monitoring network; while the data points before July 28, 2007 are the historical data obtained from Nebraska Water Center's bi-annual manual sample at the well. The higher and lower bound of the figure shows the recorded max and min groundwater level observed at the well since June, 1954. The historical average groundwater level

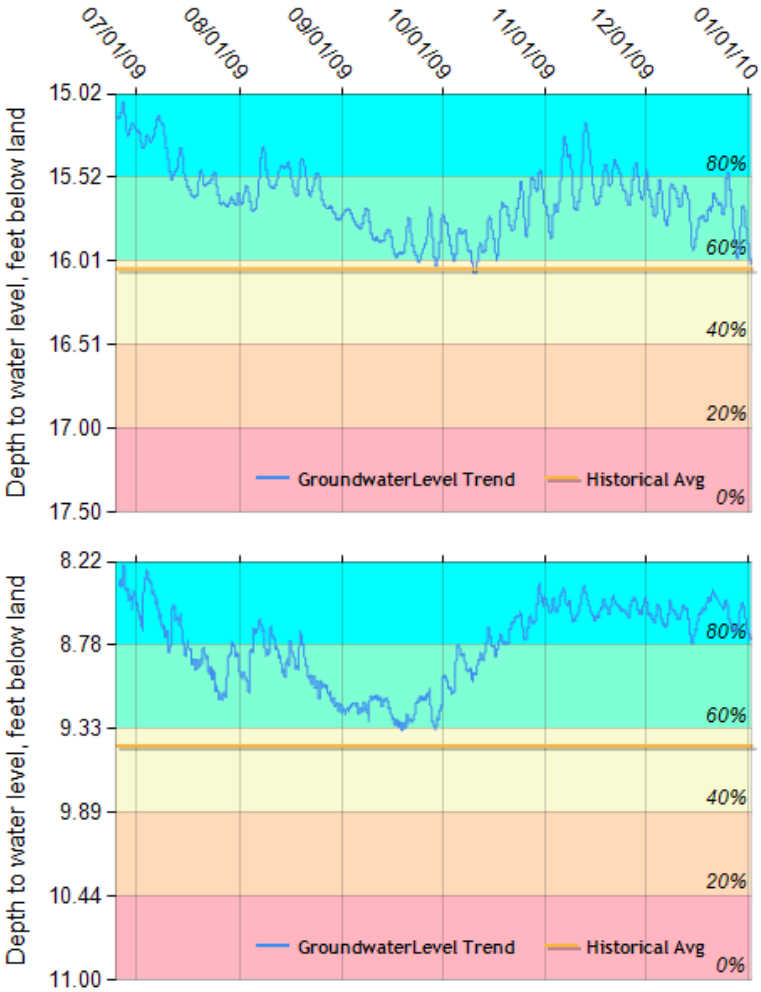


Figure 5.10: The groundwater level data collected from two adjacent observation wells located in Brown County, NE, show the highly correlated groundwater fluctuation during summer and different groundwater fluctuation during Fall, 2009.

helps the visitor understand the current drought condition of the well. Fig. 5.9 illustrate how the groundwater level responds to a 100-year storm event reported at the site in May, 2008, which cannot be observed by an bi-annual sample at the well. The results reveals that properly adapting the sensing frequency is important for efficient environment monitoring, as we discussed in Chapter 2. Since groundwater level usually fluctuate extremely slow (less than 1 feet in a month), low sensing frequency should be used in continuous monitoring application for energy and bandwidth efficiency, which is extremely important for tier

two WSNs. While a special event is detected, the sensing frequency in the hot spot should be adapted quickly to satisfy the event-specific reliability and timeliness requirements.

Fig. 5.10 shows the groundwater level curves of two nearby observation wells (about 2 mile away) located at Brown County, Nebraska. From Fig. 5.10, we observe that the groundwater level data collected from these two wells are highly correlated during summer (July 01, 2009 to September 15, 2009) but fluctuate differently during fall (September 15, 2009 to November 15, 2009). The observation based on the data reveals that increasing the node density in interesting areas can help improve the observation accuracy; however, proper data aggregation is required for timely correlated data collected by the spatially correlated sensors. In addition, the data aggregation policy should be adaptable to the varying environmental conditions.

Fig. 5.11 shows the groundwater level data collected from an observation well located in Cass County, Nebraska. The daily precipitation statistics of the same area [102] are given for comparison. Since the observation well is a shallow well under sandhills, the groundwater level responds quickly to the precipitation pikes, which is highlighted by red circles in the figure. The groundwater level data collected from this area are valuable for geoscientists to capture the aquifer recharge process and construct local hydrological model, based on which the aquifer trends can be predicted for long-term strategic planning. A tier-two WSN deployment is highly appreciated in such areas to improve the local sensing scale and variety so that a comprehensive hydrological model, including both groundwater and surface water systems, can be derived.

5.5 Summary

The real-time groundwater monitoring network discussed in this chapter illustrates how wireless communication and networking technologies help in improving the accuracy, cov-

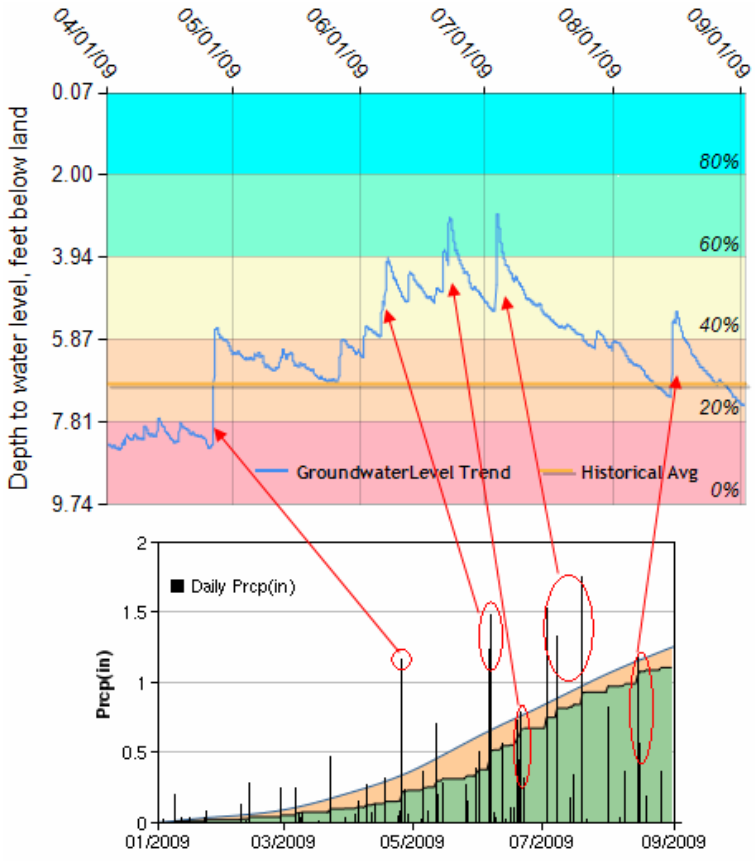


Figure 5.11: The groundwater level data collected from observation wells located under sandhills in Cass County, NE. show how groundwater level responds quickly to local precipitations.

erage, cost efficiency and re-tasking capability of traditional environment monitoring networks. We collaborated with geoscientists from School of Natural Resources Center at University of Nebraska to define the core application requirements, which presents a collection of guidelines that can serve as a basis for a general environmental monitoring network architecture for many such applications.

The real-time groundwater level data collected from the current network deployment shows that the tier-one satellite network can provide the reliable communication between the rural monitoring sites and the base station, while enabling easy failure detection and isolation in large-scale environmental monitoring networks. The power and casing system

designed for the on-site installation are proven to be feasible and reliable. The network managers can conduct the data and network management at the network base station located at University of Nebraska. The groundwater level data collected by the network can be accessed publicly through our web server (snr-1349.unl.edu), and have already been used by geoscientists in Nebraska to create the latest state groundwater map. The deployed network will be integrated with National Integrated Drought Information System (NIDIS) to derive support decisions.

Some interesting sites have be identified based on the preliminary data set collected from the network. The deployment of tier-two WSNs to provide extended and versatile sensing capability at these hot spots are highly anticipated by the geoscientists, which makes a comprehensive drought monitoring system possible by integrating not only groundwater, but also surface water, soil and weather systems. With the bi-directional GOES communication capability enabled in 2010, the remote network re-tasking, including the over-the-air reconfiguration on both the GOES access points and local WSN nodes can be implemented and tested. With the increased node density and the reduced transmission range provided by the tier-two WSNs, how to guarantee the reliability and timeliness in multi-hop wireless communication, how to provide the adaptive in-network data processing, and how to conduct network resource management will become the future focuses of the project. The complete two-tiered network will also provide us a testbed to conduct the important reliable and real-time event sensing researches with realistic monitoring applications.

5.6 Acknowledgements

This work was funded through a partnership with the U.S. Department of Agriculture Risk Management Agency. The projet, entitled ”An Integrated Real-Time Groundwater-Level

Monitoring Network to support Drought Impact Assessment and Mitigation Programs", was done in conjunction with the School of Natural Resources at the University of Nebraska. I would like to thank Dr. Mark Burbach, the PI of the project, for providing the background information on groundwater and drought monitoring systems, which has been used in this chapter. The successful deployment of the state-wide groundwater monitoring network would not be possible without Dr. Burbach's dedicated work. I also would like to thank Dr. Cody L. Knutson for his help on obtaining the GOES satellite communication channels used in this project.

Chapter 6

Conclusion and Future Work

6.1 Research on Soft QoS provisioning

In this work, we explored how to provide QoS control in event-based WSNs through multi-layer protocol design so that the application-specific soft QoS requirements for end-to-end data communication can be satisfied in terms of latency and reliability. We summarized the characteristics of WSNs and point out the major challenges of QoS provisioning for event-based sensing applications under dynamic WSN environment. Based on that, we identify soft QoS provisioning in WSNs as a cross-layer task.

6.1.1 Latency Domain

By exploiting the independencies between MAC and network layer controls, a service-differentiated real-time forwarding scheme, SDRCS, is proposed in Chapter 2. SDRCS aims at providing transmission latency guarantees on end-to-end event data convergecast. The event data collected by a group of sensor nodes covering the event area can be assigned a specific end-to-end transmission latency requirement, also called event deadline. SDRCS can prioritize the packet transmission based on the deadline requirements and the end-to-

end hop-count estimations associated with different event data packets, so that the packets that require larger traversal speed can be scheduled earlier for transmission through a prioritized CSMA/CA with RTS/CTS MAC design. Meanwhile, SDRCS utilizes an integrated MAC and network layer operation, which combines the route selection with the RTS/CTS packet exchange process. By this way, SDRCS enables receiver-contention-based packet forwarding to maximize the packet traversal speed based on local network and channel conditions. SDRCS also includes a per-hop packet traversal speed estimation component, through which the packet schedulability can be predicted based on its required and achievable traversal speed. The packet prioritization, admission control and drop policies are then applied according to the schedulability prediction. SDRCS design guarantees that 1) a packet associated with larger traversal speed requirement can be assigned higher priority level, 2) a packet with higher priority level can be scheduled for transmission earlier than other competing packets within the interference range, 3) a packet can always be forwarded to a next hop achieving the largest single hop traversal speed, and 4) all packets received by the sink are subjected to the event-based deadline requirements. According to our performance analysis, compared with existing service differentiated real-time communication schemes RAP [26] and MMSpeed [21], SDRCS is able to improve the on-time delivery ratio by about 20% for mixed priority traffic flows in WSNs with or without communication voids. SDRCS also achieves higher end-to-end communication throughput in terms of supporting higher data rates without network congestion.

Based on the SDRCS design, we find out that quantitatively characterizing the achievable end-to-end packet delivery rate using proposed receiver-contention-based forwarding (anycasting) scheme is vital for applying SDRCS into real-world event-based WSN applications. An analytical framework is thus proposed in Chapter 3 to address this problem. Using a realistic log-normal channel model, we provide a statistical end-to-end latency and energy analysis for anycasting operation, from which the probability of satisfying cer-

tain end-to-end latency and energy requirement for low-rate event traffic can be derived with a confidence level. Based on the end-to-end analysis, we provide insights on how to design latency and energy efficient forwarding metrics. In addition, for low-power duty-cycled WSNs, we investigate how the anycasting preamble length affects the achievable end-to-end latency and energy efficiency, and propose a series of preamble length control guidelines for low and extremely low duty-cycled WSNs. According to our analytical results and simulation validation, two forwarding metrics proposed based on our analytical work help reduce the end-to-end latency and energy consumption of anycasting operation with moderate preamble length. The proposed preamble length control guidelines help reduce, by more than half, the end-to-end energy and latency costs in low and extremely-low duty-cycled WSNs.

6.1.2 Reliability Domain

With fixed channel characteristics and physical layer parameters, MAC and network layer design determines the end-to-end communication capacity or throughput of a multi hop WSN. However, with increased amount of traffic being injected into the network, the end-to-end latency will be increased because of higher network contention and congestion level. In other words, the on-time packet delivery rate will be affected by the traffic volume. In Chapter 4, we focus on event-based end-to-end reliability guarantee in multi hop WSNs through transport layer traffic rate control and congestion control. First, an event sensing fidelity metric is defined based on the ratio of observed event goodput at the sink to the required event goodput specified by the sensing application. Then, through a cases study to explore the relationship between the event source rate and the achieved event sensing fidelity, we divide the network status into three regions: 1) Region 1, where the event sensing fidelity increases linearly with the increasing event source rate, and the packet loss rate only due to wireless link error; 2) Region 2, where the required event source rate for a spe-

cific event sensing fidelity is not predictably linear, and the packet loss rate is dynamically increased because of higher congestion level; and 3) Region 3, where the event sensing fidelity cannot be increased by increasing event source rate, and the packet loss rate is dramatically increased because of full congestion observed at all event nodes. Based on the observations obtained from the case study, a transport layer Loss-Tolerant Reliable Event Sensing (LTRES) protocol, is proposed for continuous surveillance application with multiple event areas. According to the identified network status, LTRES performs distributed source rate adaptation at event nodes with loss rate based congestion control mechanism. By this way, the event sensing fidelity requirements can be satisfied under certain network capacity. An equation based fair rate control algorithm is provided to improve the fairness among the traffic flows sharing the congestion path. The performance evaluations show that, compared with existing reliable event transport protocol ESRT [64], LTRES provides both reliable data transport for sustainable LTR requirements and best-effort data transport services for unsustainable LTR requirements. It achieves faster convergence time, lower packet loss rate and better bandwidth utilization, especially for sensing applications with high level of fidelity requirements.

6.2 Real-World Sensor Network Application

In Chapter 5, we describe the design, implementation and deployment details of a two-tier real-time environmental monitoring network in Nebraska. Our state-wide sensor network infrastructure uses WSN technology in tier-two networks to conduct dynamic sensing tasks with high resolution and flexibility. The satellite communication technology is used in tier-one back-bone network to provide reliable and low-cost long-haul connectivity between each local WSN and the central base station. The proposed two-tier sensor network infrastructure demonstrates how the wireless communication and networking technologies

help in improving the accuracy, flexibility, and cost efficiency of large-scale real world monitoring applications. Currently, the entire tier-one infrastructure has been designed and deployed to provide state-wide wireless conductivities for 54 monitoring sites equipped with 1-4 water level transducers. The real-time groundwater level data collected from the current network deployment show that the tier-one satellite network can provide the reliable communication between the rural monitoring sites and the base station, while enabling easy failure detection and isolation in large-scale environmental monitoring networks. The power and casing system designed for the on-site installation are proven to be feasible and reliable. The network managers can conduct the data and network management at the central base station located at University of Nebraska. The groundwater level data collected by the network can be accessed publicly through our web server (snr-1349.unl.edu), and have already been used by geoscientists in Nebraska to create the latest state groundwater map.

Some interesting sites have been identified based on the preliminary data set collected from the network. The on-site WSN deployment at these selected monitoring sites to provide improved local coverage and versatile sensing capability is planned in the near future. The communication protocol design for tier-two WSN has been conducted through our theoretical research discussed in Chapter 2 to Chapter 4, and the deployment feasibility research is given in Section 5.3. With the complete deployment of our two-tier network architecture, the state-wide network infrastructure will serve as a real-world sensor network testbed for large-scale environmental monitoring applications.

6.3 Future Work

Some future work that could enrich and extend the work described in the dissertation is summarized in this section. First, in terms of cross-layer protocol design, how to characterize the essential information that should be exchanged across layers and be used in protocol

operation needs to be addressed. Currently, SDRCS and LTRES focus on latency and reliability control through separate layer controls, where the cross-layer information exchange is limited. For example, SDRCS enables link quality and capacity information to be coupled with network layer routing decisions. It also enables link layer resource allocation decisions to be coupled with application-specific traffic prioritization level. However, how the transport layer data rate information could affect the per-hop link and network layer decision is not considered in our design. If the prioritized forwarding scheme adopted by SDRCS could consider the amount of outgoing traffic at nearby sensor nodes and adapt the scheduling and routing decision according to that, an improvement on the traffic distribution and congestion level of the network can be expected. In addition, the cross-layer interaction between physical layer controls, such as the transmission power, receiving sensitivity and residual energy level, and higher layer controls, such as prioritized packet scheduling policy, routing decision and event source rate adaptation need to be considered.

Second, in terms of design complexity and performance gain tradeoff, how to improve the overall protocol performance while retaining the distributed and modular features of the design need to be addressed. For example, SDRCS makes forwarding decision in a fully distributed manner, where the receiver-contention-based forwarding only requires local channel or topology knowledge within one hop. In contrast, some other real-time routing schemes, such as [59, 103], utilize control packet exchange among two or more hops to obtain higher level of global knowledge, through which the forwarding decision can be better optimized. In this case, dealing with the tradeoff between operation complexity (control overhead) and the overall performance to optimize the protocol design becomes important for QoS provisioning in highly resource constrained WSNs. On the other hand, from a cross-layer integration perspective, although many studies have demonstrated that significant performance gain can be achieved, cross-layer design increases the design complexity and diminishes the advantages of modularity, which may in turn create unintentional inter-

actions between layers and lead to undesirable consequences on the stability of the system [104]. In fact, protocol layers are extremely useful in allowing designers to optimize a single protocol layer design without the complexity and expertise associated with considering other layers. Keeping some form of separation, while allowing layers to actively interact, appears to be a good compromise for enabling interaction between layers without eliminating the layering principle [16].

Third, in terms of QoS provisioning for WSNs with multimedia streaming traffic, dealing with the data correlation and the network jitter is required in addition to the basic event sensing fidelity control proposed in this work. The temporal and spatial correlation among the event data need to be better resolved so that minimum amount of aggregated streaming traffic can be transmitted end-to-end for better bandwidth and energy efficiency. The streaming traffic need to be time-stamped, which means network synchronization is necessary. Both required data rate and required jitter rate need to be achieved at the sink as the end-to-end reliability metric, so that the streaming quality can be guaranteed. Proper jitter buffer control algorithms need to be developed at the sink to determine whether the data arrived on schedule and to adjust the buffer depth to accommodate packet reordering. In addition, providing better admission control, traffic scheduling, and bandwidth allocation algorithm for dynamic streaming traffic is vital to improve the overall QoS utility and fairness level achieved by the end-to-end communication protocol.

Fourth, in terms of performance analysis, a more comprehensive analytical model to statistically evaluate the end-to-end performance of anycasting operation in multi hop WSNs can facilitate the QoS provisioning research and its real-world applications. Currently, the proposed end-to-end energy and latency model only characterizes the performance of receiver-contention based forwarding operation adopted by SDRCS design under low traffic rate. The performance of other variations of anycasting using different contention scenarios, such as those proposed in [59, 55], can be studied through a modification to our

existing analytical framework. As a result, a comparative study of different anycasting contention scenarios can be conducted to facilitate better protocol design. For multimedia streaming WSN applications, providing a queuing model to the existing analytical framework is also important for modeling the end-to-end anycasting performance with different traffic patterns and traffic rates. In addition to the performance modeling, evaluating the proposed protocol under WSN test-bed with noisy and fading channel and real-world event traffic is important for protocol design validation. Implementing a complete layer stack in a WSN test-bed and observing the end-to-end performance through case studies can also help us identify the essential information that should be exchanged across layers and facilitate better cross-layer design.

Lastly, in terms of our state-wide sensor network infrastructure, the tier-two multi-hop WSN need to be deployed at the selected monitoring sites to make he comprehensive drought monitoring system possible. Based on the two-tier network infrastructure, different WSN communication protocols can be implemented and tested through remote retasking for real-world sensing applications in outdoor environment, while the sensing data and network performance can be analyzed at the base station. The deployed infrastructure will help expedite the commercial adoption of the WSN protocol design. In addition, a series of new research challenges raised in large-scale tiered sensor network architecture, such as cross-tier topology control, data aggregation, QoS control, and software architecture design, can be studied using the deployed infrastructure. Specifically, the topology control focuses on improving the coverage of the network with optimized tier-one node placement, optimized duty cycle control and minimized energy consumption. The data aggregation is an important task for the tier-one node to minimize the traffic and improve the reliability. QoS control also needs to be addressed in a cross-tier manner so that guaranteed communication services can be provided within different layers that use different wireless technologies, and cross-layer through proper switching interface. The software architec-

ture for tiered network also raises new problems, such as the support of efficient queries, network re-tasking, network addressability, and network failure detection, over multiple tier-two WSNs.

Bibliography

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks Journal (Elsevier)*, vol. 38, pp. 393–422, 2002.

[2] J. Polastre, R. Szewczyk, A. Mainwaring, D. Culler, and J. Anderson, "Analysis of wireless sensor networks for habitat monitoring," *Wireless Sensor Networks*, pp. 399–423, 2004.

[3] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo *et al.*, "Deploying a wireless sensor network on an active volcano," *IEEE Internet Computing*, vol. 10, no. 2, pp. 18–25, March 2006.

[4] C. Sharp, S. Schaffert, A. Woo, N. Sastry *et al.*, "Design and implementation of a sensor network system for vehicle tracking and autonomous interception," in *Proc. of EWSN 2005: The Second European Workshop on Wireless Sensor Networks*, January 2005, pp. 93–107.

[5] J. Tavares, F. J. Velez, and J. M. Ferro, "Application of wireless sensor networks to automobiles," *Measurement Science Review*, vol. 8, no. 3, 2008.

[6] S. Oh, L. Schenato, P. Chen, and S. Sastry, "Tracking and coordination of multiple agents using sensor networks: System design, algorithms and experiments," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 234–254, Jan 2007.

[7] J. R. Moyne and D. M. Tilbury, "The emergence of industrial control networks for manufacturing control, diagnostics, and safety data," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 29–47, Jan 2007.

[8] T. Gao, C. Pesto, L. Selavo, Y. Chen *et al.*, "Wireless medical sensor networks in emergency response: Implementation and pilot results," in *Proc. of 2008 IEEE International Conference on Technologies for Homeland Security*, Waltham, MA, May 2008.

[9] S. Patel, K. Lorincz, R. Hughes, N. Huggins *et al.*, "Monitoring motor fluctuations in patients with parkinson's disease using wearable sensors," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 6, November 2009.

[10] S. N. Pakzad, G. L. Fenves, S. Kim, and D. E. Culler, "Design and implementation of scalable wireless sensor network for structural monitoring," *Journal of Infrastructure Engineering*, vol. 14, no. 1, pp. 89–101, March 2008.

[11] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks Journal (Elsevier)*, vol. 52, no. 12, pp. 2292–2330, 2008.

[12] M. Zuniga and B. Krishnamachari, "An analysis of unreliability and asymmetry in low-power wireless links," *ACM Transactions on Sensor Networks*, vol. 3, no. 2, p. 7, June 2007.

[13] G. Werner-Allen, P. Swieskowski, and M. Welsh, "Motelab: a wireless sensor network testbed," in *Proc. of IPSN '05: the 4th international symposium on Information processing in sensor networks*, Los Angeles, CA, 2005, p. 68.

[14] WUSTL, "The WUSTL wireless sensor network testbed," August 2008. [Online]. Available: http://www.cse.wustl.edu/wsn/index.php?title=Testbed

[15] UMelb, "BigNet sensor network testbed," September 2009. [Online]. Available: http://www.issnip.unimelb.edu.au/project-item/bignet_testbed/bignet_sensor_network_testbed

[16] Q. Zhang and Y.-Q. Zhang, "Cross-layer design for QoS support in multihop wireless networks," *Proceedings of the IEEE*, vol. 96, no. 1, pp. 64–76, January 2008.

[17] Y. Xue, B. Ramamurthy, and M. C. Vuran, "A service-differentiated real-time communication scheme for wireless sensor networks," in *Proc. of SenseApp '08: The Third IEEE International Workshop on Practical Issues in Building Sensor Network Applications*, 2008, pp. 748–755.

[18] Y. Xue, B. Ramamurthy, and Y. Lu, "A distributed reliable data transport strategy for event based wireless sensor networks," in *Proc. of SenSys '06: The 4th ACM Conference on Embedded Networked Sensor Systems*, Boulder, CO, 2006, pp. 407–408.

[19] Y. Xue, B. Ramamurthy, and Y. Wang, "Providing reliable data transport for dynamic event sensing in wireless sensor networks," in *Proc. of ICC'08: IEEE International Conference on Communications*, Beijing, China, May 2008, pp. 3146–3150.

[20] ——, "LTRES: A loss-tolerant reliable event sensing protocol for wireless sensor networks," *Computer Communications*, vol. 32, pp. 1666–1676, 2009.

[21] E. Felemban, C. Lee, and E. Ekici, "MMSPEED: Multipath multi-speed protocol for qos guarantee of reliability and timeliness in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 738–754, June 2006.

[22] Y. Xue, B. Ramamurthy, and M. C. Vuran, "Cost efficiency of anycast-based forwarding in duty-cycled wsns with lossy channel," *under review*, March 2010.

[23] T. He, P. Vicaire, T. Yan, L. Luo *et al.*, "Achieving real-time target tracking usingwireless sensor networks," in *Proc. of the IEEE RTAS'06: IEEE Real-Time and Embedded Technology and Applications Symp.*, San Jose, California, April 2006, pp. 37–48.

[24] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Computer Networks Journal (Elsevier)*, vol. 51, no. 4, pp. 921–960, 2007.

[25] O. Chipara, Z. He, G. Xing, Q. Chen *et al.*, "Real-time power-aware routing in sensor networks," in *Proc. of IWQoS 2006: The 14th IEEE International Workshop on Quality of Service*, New Haven, CT, June 2006, pp. 83–92.

[26] C. Lu, B. Blum, T. Abdelzaher, J. Stankovic, and T. He, "RAP: A real-time communication architecture for large-scale wireless sensor networks," in *Proc. of IEEE RTAS 2002: IEEE Real-Time and Embedded Technology and Applications Symp.*, September 2002.

[27] T. He, J. Stankovic, C. Lu, and T. Abdelzaher, "A spatiotemporal communication protocol for wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 16, pp. 995–1006, May 2005.

[28] I. Aad and C. Castelluccia, "Differentiation mechanisms for IEEE 802.11," in *Proc. of IEEE INFOCOM 2001*, vol. 1, Anchorage, AL, April 2001, pp. 209–218.

[29] Y. Tian, A. Dogan, and F. Özgüner, "An urgency-based prioritized mac layer protocol for real-time traffic in ad-hoc wireless networks," in *Proc. of the 17th International Symposium on Parallel and Distributed Processing*, Washington, USA, April 2003.

[30] I. Rhee, A. Warrier, J. Min, and L. Xu, "DRAND: distributed randomized TDMA scheduling for wireless ad-hoc networks," in *Proc. of MobiHoc 2006: the 7th ACM*

international symposium on Mobile ad hoc networking and computing, New York, NY, USA, May 2006, pp. 190–201.

[31] B. D. Bui, R. Pellizzoni, M. Caccamo, C. F. Cheah, and A. Tzakis, "Soft real-time chains for multi-hop wireless adhoc networks," in *Proc. of IEEE RTAS 2007: IEEE Real-Time and Embedded Technology and Applications Symp.*, April 2007.

[32] IEEE802.11WG, "Draft supplement to IEEE standard802.11-1999: Medium access control (MAC) enhancements for quality of service (QoS)," 2003.

[33] M. Zorzi and R. Rao, "Geographic random forwarding (GeRaf) for ad hoc and sensor networks: multihop performance," *IEEE Transactions on Mobile Computing*, no. 4, December 2003.

[34] I. F. Akyildiz, M. C. Vuran, and O. B. Akan, "A cross layer protocol for wireless sensor networks," in *Proc. of IEEE CISS '06: IEEE Conference on Information Sciences and Systems*, Princeton, NJ, March 2006.

[35] P. Casari, M. Nati, C. Petrioli, and M. Zorzi, "ALBA: An adaptive load-balanced algorithm for geographic forwarding in wireless sensor networks," in *Proc. of IEEE MILCOM 2006*, Washington, DC, September 2006.

[36] P. Casari, M. Nati, C. Petrioli *et al.*, "Efficient non-planar routing around dead ends in sparse topologies using random forwarding," in *Proc. of IEEE ICC '07*, Glasgow, Scotland, June 2007.

[37] L. Galluccio, A. Leonardi, G. Morabito, and S. Palazzo, "A mac/routing cross-layer approach to geographic forwarding in wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 6, pp. 872–884, 2007.

[38] Y. Zhu and N. L.M., "Probabilistic approach to provisioning guaranteed qos for distributed event detection," in *Proc. of the IEEE INFOCOM'08*, Phoenix, AZ, April 2008, pp. 592–600.

[39] V. C. Gungor, O. B. Akan, and I. F. Akyildiz, "A real-time and reliable transport (RT) 2 protocol for wireless sensor and actor networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 16, no. 2, pp. 359–370, 2008.

[40] "MicaZ wireless module." [Online]. Available: http://www.xbow.com/ Products/Product_pdf_files/Wireless_pdf/MICAz_Datasheet.pdf

[41] H. Zhang, A. Arora, Y. Choi, and etc., "Reliable bursty convergecast in wireless sensor networks," *Computer Communication*, vol. 30, no. 13, pp. 2560–2576, 2007.

[42] S. Vural and E. Ekici, "Hop-distance based addressing and routing for dense sensor networks without location information," *Ad Hoc Netw.*, vol. 5, no. 4, pp. 486–503, 2007.

[43] D. Niculescu and B. Nath, "DV based positioning in ad hoc networks," *Journal of Telecommunication Systems*, vol. 22, no. 1-4, January 2003.

[44] S. Gopi, G. Kannan, D. Chander, U. Desai, and S. Merchant, "PULRP: Path unaware layered routing protocol for underwater sensor networks," in *Proc. of the IEEE ICC'08*, Beijing, China, May 2008, pp. 3141–3145.

[45] T. F. Abdelzaher, S. Prabh, and R. Kiran, "On real-time capacity limits of multihop wireless sensor networks," in *Proc. of IEEE RTSS 2004*, Lisbon, Portugal, December 2004.

[46] M. Zuniga, K. Seada, B. Krishnamachari, and A. Helmy, "Efficient geographic routing over lossy links in wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 4, no. 3, pp. 1–33, 2008.

[47] D. Chen and P. K. Varshney, "A survey of void handling techniques for geographic routing in wireless networks," *IEEE Communications Surveys and Tutorials*, vol. 9, pp. 50–67, 2007.

[48] D. Chen, J. Deng, and P. K. Varshney, "A state-free data delivery protocol for wireless sensor networks," in *Proc. of IEEE WCNC 2005*, New Orleans, LA, March 2005.

[49] ——, "On the forwardingarea of contention-based geographic forwarding for ad hoc and sensor networks," in *Proc. of IEEE SECON 2005*, Santa Clara, CA, September 2005.

[50] X. Zeng, R. Bagrodia, and M. Gerla, "GloMoSim: A library for parallel simulation of large-scale wireless networks," in *Workshop on Parallel and Distributed Simulation*, 1998, pp. 154–161.

[51] J. Kim, X. Lin, N. B. Shroff *et al.*, "Optimal anycast technique for delay-sensitive energy-constrained asynchronous sensor networks," in *Proc. of IEEE INFOCOM'09*, Rio de Janeiro, Brazil, April 2009.

[52] M. Zorzi and R. Rao, "Geographic random forwarding (GeRaf) for ad hoc and sensor networks: multihop performance," *IEEE Transactions on Mobile Computing*, no. 4, December 2003.

[53] R. C. Shah, S. Wietholter, A. Wolisz *et al.*, "Modeling and analysis of opportunistic routing in low traffic scenarios," in *Proc. of WIOPT '05*, Washington, DC, April 2005, pp. 294–304.

[54] J. Kim, X. Lin, N. B. Shroff, and P. Sinha, "On maximizing the lifetime of delay-sensitive wireless sensor networks with anycast," in *Proc. of IEEE INFOCOM'08*, Phoenix, Arizona, April 2008.

[55] S. Liu, K.-W. Fan, and P. Sinha, "CMAC: Energy efficient MAC layer design for sensor networks with anycasting," *IEEE Transactions on Sensor Networks*, March 2009.

[56] M. Zuniga, S. Karim, K. Bhaskar *et al.*, "Efficient geographic routing over lossy links in wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 4, no. 3, pp. 1–33, May 2008.

[57] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proc. of SenSys '04*, New York, NY, November 2004, pp. 95–107.

[58] Y. Xue, B. Ramamurthy, and M. C. Vuran, "A service-differentiated real-time communication scheme for wireless sensor networks," in *Proc. of IEEE SenseApp '08*, Montreal, Canada, October 2008.

[59] S. Biswas and R. Morris, "ExOR: opportunistic multi-hop routing for wireless networks," in *Proc. of SIGCOMM '05*, New York, NY, January 2005, pp. 133–144.

[60] R. Zheng and C. Li, "How good is opportunistic routing?: a reality check under Rayleigh fading channels," in *Proc. of MSWiM '08*, New York, NY, October 2008, pp. 260–267.

[61] Chipcon, "Datasheet for chipcon (TI)CC2420 2.4 ghz IEEE 802.15.4/zigbee RF transceiver," 2006. [Online]. Available: http://ti.com

[62] S. Lee, B. Bhattacharjee, and S. Banerjee, "Efficient geographic routing in multihop wireless networks," in *Proc. of MobiHoc '05*. New York, NY, USA: ACM, 2005, pp. 230–241.

[63] M. C. Vuran, O. B. Akan, and I. F. Akyildiz, "Spatio-temporal correlation: theory and applications for wireless sensor networks," *Computer Networks Journal (Elsevier)*, vol. 45, no. 3, pp. 245–259, 2004.

[64] O. B. Akan and I. F. Akyildiz, "Event-to-sink reliable transport in wireless sensor networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 13, pp. 1003–1016, 2005.

[65] J. Paek and R. Govindan, "RCRT: rate-controlled reliable transport for wireless sensor networks," in *SenSys '07: Proceedings of the 5th international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2007, pp. 305–319.

[66] C. Y. Wan, A. T. Campbell, and L. Krishnamurthy, "PSFQ: a reliable transport protocol for wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 862–872, 2005.

[67] F. Stann and J. Heidemann, "RMST: Reliable data transport in sensor networks," in *Proc. of the First International Workshop on Sensor Network Protocols and Applications*, April 2003, pp. 102–112.

[68] N. Tezcan and W. Wang, "Art: an asymmetric and reliable transport mechanism for wireless sensor networks," *International Journal of Sensor Networks*, vol. 2, no. 3/4, pp. 188–200, 2007.

[69] C. Y. Wan, S. B. Eisenman, and A. T. Campbell, "CODA: Congestion detection and avoidance in sensor networks," in *Proc. of SenSys '03: The 1st ACM Conference on Embedded Networked Sensor Systems*, Los Angeles, CA, NOV 2003, pp. 266–279.

[70] B. Hull, K. Jamieson, and H. Balakrishnan, "Mitigating congestion in wireless sensor networks," in *Proc. of SenSys '04: The 2nd ACM Conference on Embedded Networked Sensor Systems*, Baltimore, MD, USA, 2004, pp. 134–147.

[71] C. T. Ee and R. Bajcsy, "Congestion control and fairness for many-to-one routing in sensor networks," in *Proc. of SenSys '04: The 2nd ACM Conference on Embedded Networked Sensor Systems*, Baltimore, MD, USA, 2004, pp. 148–161.

[72] S. Rangwala, R. Gummadi, R. Govindan, and etc., "Interference-aware fair rate control in wireless sensor networks," in *Proc. of SIGCOMM '06*, Pisa, Italy, 2006, pp. 63–74.

[73] J. Zhu, S. Chen, B. Bensaou, and K. Hung, "Tradeoff between lifetime and rate allocation in wireless sensor networks: A cross layer approach," in *Proc. of IEEE INFOCOM 2007*, Anchorage, Alaska, USA, May 2007, pp. 267–275.

[74] O. B. Akan and I. F. Akyildiz, "ARC: the analytical rate control scheme for real-time traffic in wireless networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 12, no. 4, pp. 634–644, 2004.

[75] F. Baccelli and D. Hong, "AIMD, fairness and fractal scaling of TCP traffic," in *Proc. of IEEE INFOCOM 2002*, New York, NY, USA, June 2002.

[76] J. Kenny, N. Barber, S. Hutson, K. Linsey, J. Lovelace, and M. Maupin, "Estimated use of water in the united states in 2005," *U.S. Geological Survey Circular 1344*, p. 52, 2009.

[77] S. Anumalla, B. Ramamurthy, D. C. Gosselin, and M. Burbach, "Ground water monitoring using smart sensors," in *IEEE International Conference on Electro Information Technology*, 2005, pp. 1–6. [Online]. Available: http://dx.doi.org/10.1109/EIT.2005.1626962

[78] D. Wilhite, "Combating drought through preparedness," *Natural Resources Forum*, vol. 26, no. 4, pp. 275–285, 2002.

[79] ——, "Drought as a natural hazard: Concepts and definitions," in *Drought: A Global Assessment-Natural Hazards Disasters Series*. UK: Routledge Publishers, 2000.

[80] ——, "Drought: Understanding the hazard and reducing societal vulnerability," in *The second annual water law, policy and science conference*, April 2005.

[81] C. Taylor and W. Alley, "Ground-water-level monitoring and the importance of long-term water-level data," *U.S. Geological Survey Circular*, p. 52, 2001.

[82] W. Wendland, "Temporal responses of surface-water and ground-water to precipitation in illinois," *Journal of the American Water Resources Association*, vol. 37, no. 3, pp. 685–693, 2001.

[83] W. G. A. Western Governors Association, "Creating a drought early warning system for the 21st century," in *First annual water law, policy and science conference*, 2004.

[84] E. Biagioni and K. Bridges, "The application of remote sensor technology to assist the recovery of rare and endangered species," *International Journal of High Performance Computing Applications*, vol. 16, no. 3, August 2002.

[85] "CORIE: A pilot environmental observation and forecasting system (EOFS) for the columbia river," January 2010. [Online]. Available: http://www.ccalmr.ogi.edu/CORIE/

[86] R. N. Handcock, D. L. Swain, G. J. Bishop-Hurley, K. P. Patison, T. Wark, P. Valencia, P. Corke, and C. J. ONeill, "Monitoring animal behaviour and environmental interactions using wireless sensor networks, gps collars and satellite remote sensing," *Sensors*, vol. 9, no. 5, pp. 3586–3603, 2009.

[87] "James san jacinto mountains reserve environmental observatory," January 2010. [Online]. Available: http://www.jamesreserve.edu/weather.php

[88] IEEE-SA, "IEEE 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications (2007 revision)," June 2007. [Online]. Available: http://standards.ieee.org/getieee802/download/802.11-2007.pdf

[89] ——, "Wireless medium access control (MAC) and physical layer (PHY) specifications for low rate wireless personal area networks (LR-WPANs)," June 2006. [Online]. Available: http://standards.ieee.org/getieee802/download/802.15.4-2006.pdf

[90] M. Prior-Jones, "Satellite communications systems buyers' guide," Sep 2008. [Online]. Available: http://www.scor-int.org/ Working_Groups/satellite-systems-buyers.pdf

[91] IEEE-SA, "IEEE 802.16 wireless MAN standard for wireless metropolitan area networks," May 2004. [Online]. Available: http://standards.ieee.org/getieee802/download/802.16-2004.pdf

[92] M. J. Nestlebush, "The geostationary operational environmental satellite data collection system," March 1994. [Online]. Available: http://noaasis.noaa.gov/DCS/docs/goesdcs.pdf

[93] Sutron-Corporation, "GOES satellite capabilities," January 2010. [Online]. Available: http://www.sutron.com/SatelliteCommunications.html

[94] ——, "Satlink2 logger & transmitter operations & maintenance manual," October 2009. [Online]. Available: http://www.sutron.com/downloads/DownloadsUpdates/satlink_2_user_manual.pdf

[95] I. Mitretek Systems, "GOES data collection system (DCS) characterization reportm," June 1998. [Online]. Available: http://noaasis.noaa.gov/DCS/docs/dcsreport.pdf

[96] Crossbow-Technology, "IRIS OEM module datasheet," 2009. [Online]. Available: http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/IRIS_OEM_Datasheet.pdf

[97] ——, "MOTEWORKS software platform," 2009. [Online]. Available: http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MoteWorks_OEM_Edition.pdf

[98] ——, "Mib510 serial gateway datasheet," 2009. [Online]. Available: http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MIB510CA_Datasheet.pdf

[99] UNL, "Real-time groundwater monitoring network," 2009. [Online]. Available: http://snr-1349.unl.edu/

[100] I. Ilex-Engineering, "DCP data service (DDS) protocol specification," June 2003. [Online]. Available: http://dcs.noaa.gov/lrgs/DCP-Data-Service-2.2.pdf

[101] ——, "DECODES device conversion and delivery system V7.3 users guide," April 2007. [Online]. Available: http://dcs.noaa.gov/lrgs/DECODES-UserGuide-7.3.pdf

[102] H. P. R. C. Center, "Weather observation stations," 2009. [Online]. Available: http://www.hprcc.unl.edu/stations/index.php

[103] Y. Li, C. S. Chen, Y.-Q. Song, Z. Wang *et al.*, "Enhancing real-time delivery in wireless sensor networks with two-hop information," *IEEE Transactions on Industrial Informations*, vol. 5, no. 2, p. 113, May 2009.

[104] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1452–1463.